TK!Solver™ Program Reference Manual

Software Arts™

P/N 200-032

Copyright © 1983 Software Arts, Inc.
All rights reserved.

Software Arts
27 Mica Lane
Wellesley, MA 02181

Printed in the U.S.A.

TK, TK!, TK!Solver, and TK!SolverPack are all trademarks of Software Arts, Inc.

Software Arts is a trademark of Software Arts, Inc. and Software Arts Products Corp.

The TK!Solver program is a product of Software Arts, Inc. which is solely responsible for its contents.

Table of Contents

Contents i
Manual Summary i
Manual Conventions iii
Alphabetic Cross-reference v
Functional Cross-reference viii

Part I: Program Overview I-1
Program Startup I-1
TK!Solver Screen I-2
Movement Through the Program I-10
Program Environments I-12
Field Types I-16
Error Conditions I-22

Part II: Sheets II-1
Variable Sheet (V) II-3
Variable Subsheets (V>) II-8
Rule Sheet (R) II-11
Unit Sheet (U) II-13
Global Sheet (G) II-16
List Sheet (L) II-23
List Subsheets (L>) II-26
User Function Sheet (F) II-30
User Function Subsheets (F>) II-34
Plot Sheet (P) II-38
Table Sheet (T) II-42

Part III: Commands III-1
Left Arrow (⬅) III-3
Right Arrow (➡) III-3
Up Arrow (⬆) III-3
Down Arrow (⬇) III-3
Enter (◆) III-4
Break (⊗) III-5
Backspace (◀) III-5
Action (!) III-6
Blank (/B) III-10
Copy (/C) III-12
Delete (/D) III-14
Dive (>) III-16
Edit (◪) III-18

Edit Field (/E) III-21
Goto (:) III-23
Help (?) III-26
Insert (/I) III-29
List (/L) III-30
Move (/M) III-34
Print (/P) III-36
Quit (/Q) III-38
Reset (/R) III-39
Return (<) III-41
Select (=) III-42
Solve (/!) III-44
Storage (/S) III-45
Switch (;) III-51
Window (/W) III-52

Part IV: Mathematics IV-1
Equation Components IV-1
Expression Evaluation IV-6
Model Solution IV-7
Model Development Overview IV-13
Model Development Using Sheets IV-14
Model Development Using TK!Solver Features IV-16

Part V: Built-In Functions V-1

Part VI: Hardware Reference VI-1

Appendices A-1
TK!Solver Files A-1
Unit Name Conventions A-4
Prompt Messages A-7
Status Messages A-17
Error Messages A-22

Glossary G-1

Index Index-1

Manual Summary

The TK!Solver™ Reference Manual contains ten divisions: Contents, Program Overview, Sheets, Commands, Mathematics, Built-in Functions, Hardware Reference, Appendices, Glossary, and Index. The following summary indicates the topics discussed in each section.

Contents

An introduction to the TK!Solver Reference Manual, including Table of Contents, Manual Summary, Manual Conventions, and Alphabetic and Functional Cross-references to program features.

Part I: Program Overview

A general description of the program and its features, including Startup, Screen, Movement, Program Environments, Field Types, and Error Conditions.

Part II: Sheets

A detailed description of the Sheets and Subsheets in the program.

Part III: Commands

A detailed description of the Special Keys and Commands in the program.

Part IV: Mathematics

A detailed discussion of the Mathematical features of the program, including Equation Components, Expression Evaluation, Model Solution, and Model Development.

Part V: Built-in Functions

A detailed list of the Built-in Functions available with the program.

Part VI: Hardware Reference

A detailed description of the parts of the program that are computer dependent, including hardware requirements, keyboard, disk drives, printers, system notes, file specifications, global settings, printer setup strings, and size constraints.

Appendices

Additional information about the TK!Solver program, including TK!Solver Files, Unit Name Conventions, Prompt Messages, Status Messages, and Error Messages.

Glossary

A definition of terms used in all of the TK!Solver documentation.

Index

An alphabetical index to the Reference Manual.

Manual Conventions

Several conventions are used in the TK!Solver Reference Manual to identify special kinds of text:

- Important terms are indicated in boldface the first time they are used.
- The typeface "OCR-B" is used to indicate exact program output:

```
This sentence is set in OCR-B.
```

- A class of program output or user input is indicated in italic.
- Variable names and list names are indicated in italic.

Special conventions have been used to refer to the use of keys that vary from keyboard to keyboard. These keys, or combinations of keys, are represented by both symbols and names in the manuals: the symbol is listed below on the left; the name is on the right. See the Hardware Reference section for which key or combination of keys to use on your keyboard.

⬅	Left Arrow
➡	Right Arrow
⬆	Up Arrow
⬇	Down Arrow
◆	Enter
⊗	Break
◀	Backspace
◪	Edit

User input in the Reference Manual is usually indicated in uppercase. On computers that allow uppercase and lowercase, either case may be used. Program output is described as it appears on computers using both cases.

Manual Conventions

This manual is described for video screens that display 80 characters wide by 24 lines high. Some adjustments are made by the program to accommodate for other size screens, but these changes are not described within this manual since they do not greatly affect use of the program other than error messages and the Help Facility text. However, depending on the computer used, screen pictures in the manuals may differ from what appears on the user's screen.

Program data referred to as entered by the user throughout the documentation can also be loaded into the program from storage files.

Alphabetic Cross-reference

An alphabetic index of commands and sheets:

Commands

Action (!)	Performs the action appropriate to the current sheet, including model solution, list value fill-in, plot or table display.	**III-6**
Blank (/B)	Erases the current field item, or a column of items beginning with the current item.	**III-10**
Copy (/C)	Copies the current field item to the specified destination.	**III-12**
Delete (/D)	Deletes the current row, or a block of rows beginning with the current row.	**III-14**
Dive (⟩)	Displays a dependent subsheet for the current sheet or subsheet in the current window.	**III-16**
Down Arrow (⬇)	Moves the cursor down one row.	**III-3**
Edit (☑)	Edits a user response on the Prompt/Error line or an unentered item in the current entry field.	**III-18**
Edit Field (/E)	Edits the entered item in the current entry field.	**III-21**
Goto (:)	Moves the cursor to the specified field on the current sheet.	**III-23**
Help (?)	Calls the Help Facility.	**III-26**
Insert (/I)	Inserts a blank row at the current row.	**III-29**
Left Arrow (⬅)	Moves the cursor left one field.	**III-3**
List (/L)	Solves a model for lists of values associated with variables, or stores or retrieves list values.	**III-30**

Move (/M)	Moves the current row to the specified destination.	**III-34**
Print (/P)	Prints the specified rows from the current sheet onto an output device or into a file.	**III-36**
Quit (/Q)	Exits the TK!Solver program.	**III-38**
Reset (/R)	Clears variable values, the current sheet, or all data from the program.	**III-39**
Return (\)	Returns from the display of a subsheet to the previous sheet or subsheet.	**III-41**
Right Arrow (➡)	Moves the cursor right one field.	**III-3**
Select (=)	Displays the specified sheet in the current window.	**III-42**
Solve (/!)	Solves the currently defined model from any sheet.	**III-44**
Storage (/S)	Saves, loads, or deletes files containing models or portions of models.	**III-45**
Switch (;)	Moves the cursor between windows.	**III-51**
Up Arrow (⬆)	Moves the cursor up one row.	**III-3**
Window (/W)	Displays a single window or splits the screen into two windows.	**III-52**

Sheets and Subsheets

Global Sheet (G)	Contains global settings and program defaults.	**II-16**
List Sheet (L)	Contains a table of all lists.	**II-23**
List Subsheet (L›)	Contains a detailed description of one list defined on the List Sheet.	**II-26**
Plot Sheet (P)	Contains information necessary to plot list values.	**II-38**
Rule Sheet (R)	Contains equations for a model.	**II-11**
Table Sheet (T)	Contains information necessary to display a table of list values.	**II-42**
Unit Sheet (U)	Contains definitions for conversions between units of measurement.	**II-13**
User Function Sheet (F)	Contains a table of all functions defined by the user.	**II-30**
User Function Subsheet (F›)	Contains a detailed description of one function defined on the User Function Sheet.	**II-34**
Variable Sheet (V)	Contains data describing the variables of a model.	**II-3**
Variable Subsheet (V›)	Contains a detailed description of one variable defined on the Variable Sheet.	**II-8**

Functional Cross-reference

An index of commands and sheets arranged by their functions:

Movement Commands

Left Arrow (⬅)	Moves the cursor left one field.	**III-3**
Right Arrow (➡)	Moves the cursor right one field.	**III-3**
Up Arrow (⬆)	Moves the cursor up one row.	**III-3**
Down Arrow (⬇)	Moves the cursor down one row.	**III-3**
Goto (:)	Moves the cursor to the specified field on the current sheet.	**III-23**
Select (=)	Displays the specified sheet in the current window.	**III-42**
Switch (;)	Moves the cursor between windows.	**III-51**
Dive (⟩)	Displays a dependent subsheet for the current sheet or subsheet in the current window.	**III-16**
Return (⟨)	Returns from the display of a subsheet to the previous sheet or subsheet.	**III-41**
Window (/W)	Displays a single window or splits the screen into two windows.	**III-52**

Modification Commands

Move (/M)	Moves the current row to the specified destination.	**III-34**
Insert (/I)	Inserts a blank row at the current row.	**III-29**
Delete (/D)	Deletes the current row, or a block of rows beginning with the current row.	**III-14**

Copy (/C)	Copies the current field item to the specified destination.	**III-12**
Blank (/B)	Erases the current field item, or a column of items beginning with the current item.	**III-10**
Edit Field (/E)	Edits the entered item in the current entry field.	**III-21**
Edit (◪)	Edits a user response on the Prompt/Error line or an unentered item in the current entry field.	**III-18**
Reset (/R)	Clears variable values, the current sheet, or all data from the program.	**III-39**

Solution Commands

Action (!)	Performs the action appropriate to the current sheet, including model solution, list value fill-in, plot or table display.	**III-6**
Solve (/!)	Solves the currently defined model from any sheet.	**III-44**
List (/L)	Solves a model for lists of values associated with variables, or stores or retrieves list values.	**III-30**

General Program Commands

Quit (/Q)	Exits the TK!Solver program.	**III-38**
Help (?)	Calls the Help Facility.	**III-26**
Storage (/S)	Saves, loads, or deletes files containing models or portions of models.	**III-45**
Print (/P)	Prints the specified rows from the current sheet onto an output device or into a file.	**III-36**

Sheets

Variable Sheet (V)	Contains data describing the variables of a model.	**II-3**
Rule Sheet (R)	Contains equations for a model.	**II-11**
Unit Sheet (U)	Contains definitions for conversions between units of measurement.	**II-13**
Global Sheet (G)	Contains global settings and program defaults.	**II-16**
List Sheet (L)	Contains a table of all lists.	**II-23**
User Function Sheet (F)	Contains a table of all functions defined by the user.	**II-30**
Plot Sheet (P)	Contains information necessary to plot list values.	**II-38**
Table Sheet (T)	Contains information necessary to display a table of list values.	**II-42**

Subsheets

Variable Subsheet (V›)	Contains a detailed description of one variable defined on the Variable Sheet.	**II-8**
List Subsheet (L›)	Contains a detailed description of one list defined on the List Sheet.	**II-26**
User Function Subsheet (F›)	Contains a detailed description of one function defined on the User Function Sheet.	**II-34**

Program Overview

Program Startup

The way to start the TK!Solver program depends on the particular computer and operating system being used. The startup for your system is described in the TK!Solver Instruction Manual and in the Hardware Reference section of this manual.

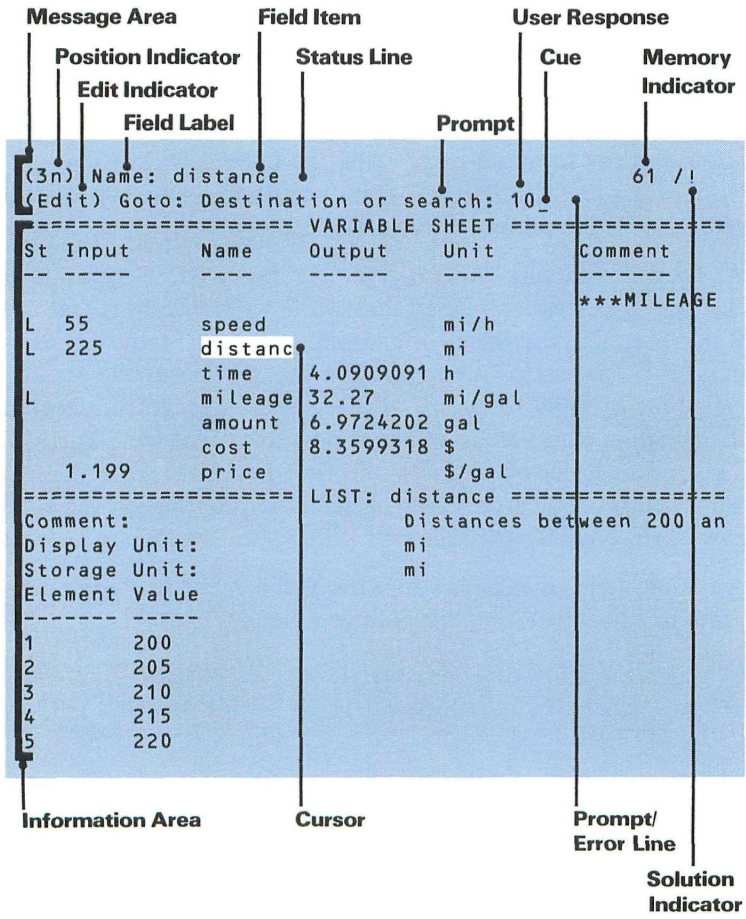
The TK!Solver program first displays a full-screen startup message with a copyright notice and program version numbers. Initiate use of the program by pressing the Enter key. The message "Loading the TK!Solver program" appears at the top of the screen while the program is being loaded.

At program startup, two windows are displayed. They contain the two most frequently used sheets: the Variable Sheet, and the Rule Sheet. The top line on the screen, called the Status line, displays program status messages. The second line of the screen, called the Prompt/Error line, displays user messages and begins the program with a message about how to call the Help Facility. This message disappears with the first keystroke. The cursor, which marks the current field, begins the program on the first row of the Rule Sheet in the first Rule field.

Any time after this display appears, you can either use the program or exit it by typing the Quit command (/QY).

The TK!Solver Screen

The TK!Solver program screen is divided into two areas: the Message Area and the Information Area.



Message Area

The Message Area is used by the program to provide status information. It consists of two lines: the Status line and the Prompt/ Error line.

Status Line
The Status line provides five kinds of status information:

The **Position Indicator** is a number and letter (or only a letter) in parentheses that indicates the address of the field marked by the cursor and changes as the cursor moves. The number, if any, represents the row; the letter stands for the label of the field. (Each field in the program has a letter name used in referring to its address.)

At program startup, **(1r)** is displayed, indicating the Rule field on the first row of the Rule Sheet.

The **Field label** on the Status line displays the label of the current field.

At program startup, it displays **Rule:**.

The **Field item** on the Status line displays the contents of the field currently marked by the cursor. Field items that are longer than the screen fields they occupy can be viewed more fully on the Status line. This includes the full display of value items that are rounded off or expressed in scientific notation so that they can display within their fields. The end of a field item even longer than the space available on the Status line can only be viewed by using the Editor on the current field and moving the cue to the end of the item.

The explanation for the letters and characters displayed in Status fields on the Variable and Rule sheets can be viewed on the Status line by moving the cursor to the field in question and looking at the complete definition on the Status line.

At program startup, no field item is displayed.

The **Memory Indicator** is a number followed by a slash that indicates the amount of remaining memory and is displayed in K (1024) bytes. When the remaining memory is less than 1 K, the Memory Indicator displays **LOW**. Only minimal actions are allowed: some actions requiring additional memory fail. When there is no remaining memory, the Memory Indicator displays **OUT**. At this point, most actions except saving a file fail, until something is deleted.

At program startup, a number is displayed representing the amount of memory available for use after the program is loaded.

The **Solution Indicator** is an exclamation point (!) that indicates the solution status of the model. It appears whenever a modification is made that may invalidate the current solution. It disappears when the Solve command is typed, any equation in the model is satisfied, and there are no errors in the model.

Prompt/Error Line

The Prompt/Error line displays the prompts, status messages, and error messages provided by the program. Certain prompts require the user to type on the Prompt/Error line.

At program startup, the Prompt/Error line displays "For Help, type ?". The Help reminder disappears when the first key is pressed.

The Prompt/Error line displays six kinds of information:

The **Edit Indicator** is the word "Edit" in parentheses that appears whenever the user is in the Editor on the current field or on the Prompt/Error line itself. It disappears when the Enter key completes the entry or the Break key cancels it.

Prompts are produced by the program to indicate the next user action. Most prompts offer a selection of options; some require entering information on the Prompt/Error line. (See the appendix on Prompts for a listing of each prompt in the program.)

The **Pointer Indicator** is a number and letter (or letter only) in parentheses that appears at the end of prompts that request the user to point to another field. It supplies the current cursor position during pointing and changes whenever the Arrow keys or the Goto command (:) move the cursor. While the Pointer Indicator is being used, the Position Indicator does not change. The Pointer Indicator disappears when the Enter key completes the command.

User responses are required by certain commands. Some commands offer a choice of options. If an option is typed, it does not appear on the Prompt/Error line. Some commands require entries to be made on the Prompt/Error line itself. The prompts for these commands are ended by a colon. The cue appears on the Prompt/Error line indicating where to type. Any information

typed on the Prompt/Error line must be completed with the Enter key. A user response that has not yet been completed by the Enter key is called an "entry" in this manual.

The Editor can be called to edit a partial entry on the Prompt/Error line by using the Edit command ⊠ any time before the final Enter key is pressed.

Status messages are displayed during certain program processes that require time to perform, such as using the solvers, or deleting, loading, or saving files. (See the appendix on Status Messages for a listing of each status message in the program.)

Error messages are produced by the program for a variety of reasons, including output errors or attempts to enter an invalid command, entry, or option. They are usually accompanied by a computer error signal.

Entry errors automatically place the user in the Editor and disappear when a valid item is entered. Command and option error messages disappear with the next valid keystroke.

See the appendix on Error Messages for a listing of each error message in the program.

Information Area

The Information Area is the part of the screen that displays the sheets on which to set up models and generate solutions.

The Information Area can display either one or two **windows** at once, allowing a view of two sheets simultaneously, or a full screen view of a single sheet. Two different views of the same sheet may also be displayed at one time; any change made to one view of a sheet is automatically made on the other. Changes made to fields that correspond on more than one sheet, such as a field on both the Variable Sheet and one of its subsheets, are made simultaneously on both sheets whether or not both are currently displayed.

Sheets are locations, viewed through windows, for entering and displaying the data for a model. There are eight main sheets:

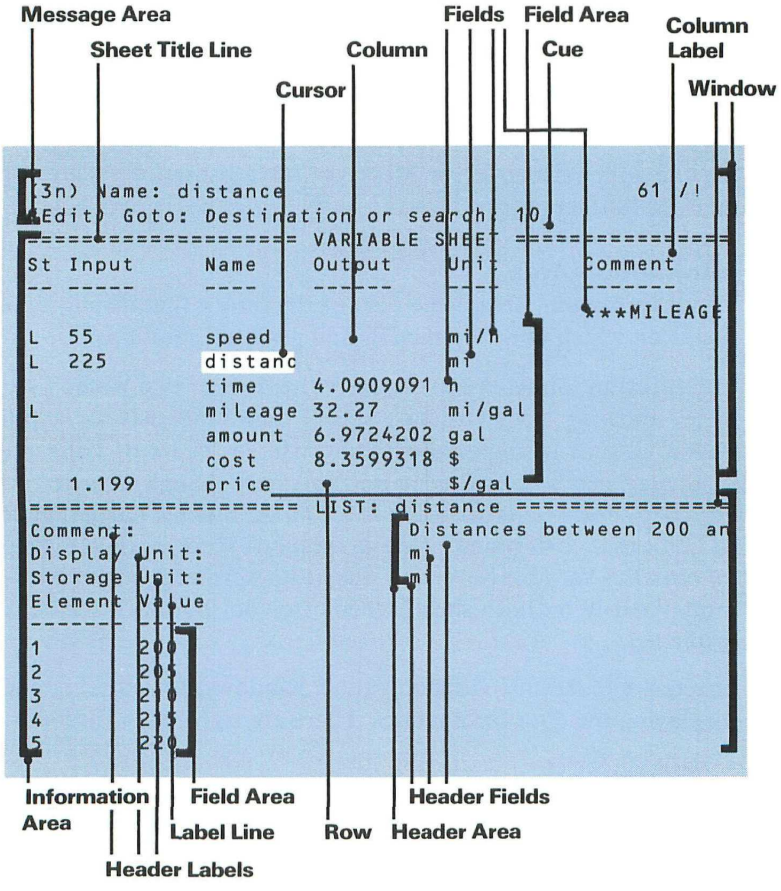
- Variable Sheet
- Rule Sheet
- Unit Sheet

Screen

- Global Sheet
- List Sheet
- User Function Sheet
- Plot Sheet
- Table Sheet

Subsheets are related to particular sheets and provide detailed information about entities on the corresponding sheet. A subsheet can be reached by placing the cursor on a row on a sheet and diving into the subsheet related to that row. (See the Dive command for more information.) There are three kinds of subsheets:

- Variable subsheets
- List subsheets
- User Function subsheets



The parts of a sheet or subsheet are:

Sheet Title Line
The Sheet Title line defines the top of a sheet. It contains the name of the sheet and a row of equal signs (=).

A Sheet Title line for a subsheet differs from one for a sheet in that the name of the described variable, list, or function is included following the subsheet title.

Fields and Labels
Fields are positions on a sheet where the cursor can be moved and information entered or displayed. Each field is marked by a label. There are three types of fields: entry fields, option fields, and output fields (see the section on Field Types). Each field in the program has a unique address that appears as the Position Indicator on the Status line whenever the cursor is located on that field.

Fields have a display size, which is the size of the field appearing on the screen, and an item size, which is the actual length of the contents of the field.

For convenience, the Reference Manual describes fields as being organized into one of two areas: Header Areas and Field Areas.

Header Areas contain general information about the contents of a sheet or use of the program. For example, the Header Area on the Global Sheet contains information the program needs when solving equations.

A Header Area contains a block of rows, each consisting of a Header label on the left, and a Header field on the right. Each label describes the type of information contained in only one field. A field address in a Header Area consists of a single letter, which stands for the label.

Several TK!Solver commands do not work in Header Areas. For example, rows cannot be inserted, deleted, or moved. Many entries in Header fields are required, and cannot be blanked.

The Header Area is above the Field Area when both occur on the same sheet. It disappears when a Field Area is scrolled down, reappearing when the cursor is moved back up to the Header Area.

The Global Sheet and a Variable subsheet contain only Header Areas. The Plot and Table sheets, and List and User Function

subsheets, contain Header Areas as well as Field Areas. The other sheets do not contain Header Areas.

Field Areas contain fields organized in rows under a line of column labels. One label describes every field below it. Column labels are always displayed, even if the Field Area is scrolled down. All TK!Solver commands operate in Field Areas. A field address in the Field Area is a row number and a single letter, which stands for the column label.

Rows and Columns
A row is a line of labeled fields. A column is a vertical list of fields under the same label.

Items
An item is the information typed into a field by the user or displayed there by the program. Characters typed into an entry field are only entered by pressing the Enter key or an Arrow key.

There are three types of items:

- Entry item: one or more characters typed and entered by the user into an entry field.
- Option item: a single character typed by the user into an option field, interpreted and expanded as appropriate by the program.
- Output item: information displayed by the program in an output field.

An item may be longer than the display size of the field it occupies. Maximum character item length is 200 characters; real numbers are accurate up to 10 digits; maximum integer length is dependent on your computer. (See the Hardware Reference section for more information.)

Values may be rounded off or expressed in scientific notation in order to fit into the display size of various fields. However, all calculations are performed using the full accuracy.

Items that exceed their display size may be viewed more fully on the Status line by moving the cursor to the field. Items longer than the area on the Status line can only be viewed in the Editor, by scrolling to the end of the item with the Arrow keys.

Other Displays

There are four other displays that preempt all or part of the screen in the TK!Solver program:

- The Help Facility replaces the Information Area with a display labeled "HELP" at the top, where topics about the use of the TK!Solver program can be viewed.
- Plot displays replace the entire screen with a one-page graph.
- Table displays replace the entire screen with a single- or multi-page table of list values.
- Print displays replace the entire screen with lines printed to a file or printer device by the Print command (/P).

Movement Through The Program

The TK!Solver program has two moveable pointers that mark the current point of attention: the cursor moves around the screen; the cue moves within text.

Cursor

The cursor marks the current field. Each sheet has an initial cursor position, the place marked by the cursor the first time a sheet is accessed. When a sheet is re-accessed the previous cursor position and view of the sheet are displayed. The cursor always returns to the initial cursor position on a subsheet.

The cursor is moved from field to field by the Arrow keys:

- ⬅ Left one field
- ➡ Right one field
- ⬆ Up one row
- ⬇ Down one row

It can also be moved by the Goto command (:); the Switch command (;), which moves it between windows; and as a result of some commands that require pointing to a new field.

If an attempt is made to move the cursor past the right or left side of the screen, or up past the top of a sheet or subsheet or down below its end, the computer produces an error signal. The bottom of most sheets and subsheets is defined at row 32000, but the number of items that can be entered is limited by the amount of memory available. Variable subsheets and the Global Sheet have fixed last lines.

If the cursor is moved down beyond the current screen or window display, the current sheet scrolls down a line at a time to display a new view. The view may be scrolled back up by moving the cursor past the top of the current display.

Cue

The cue marks the position within the text in the current field or on the Prompt/Error line. It creates an artificial space and marks an insertion point.

It appears whenever characters are typed or edited within a field, or when the user is prompted to type on the Prompt/Error line.

All characters typed appear to the left of the cue; the Backspace key deletes the character to the left of the cue. The cue can be moved around within the text and disappears, along with the space it creates, after the Enter key is pressed.

When editing, the cue is moved around a field or the Prompt/Error line by the Arrow keys:

- ⬅ Left one character
- ➡ Right one character
- ⬆ To the beginning of the text
- ⬇ To the end of the text

If an item is longer than the display size of the field or the display area of the Prompt/Error line, its view is changed as the cue moves. When the Enter key is pressed to enter the item in a field the cue disappears and the beginning of the item is displayed at the left edge of the field.

Program Environments

There are several environments or modes in the TK!Solver program in which particular user actions are valid. They are categorized for convenience only in the Reference Manual.

As the cursor moves around the program, it marks an entry field, an option field, or an output field. Modes in the TK!Solver program are dependent on where the cursor is positioned and what is typed. (See the section on Field Types for more information about valid field entries.)

There are three main program modes:

- Command mode: on an entry, option, or output field
- Entry mode: on an entry field
- Option mode: on an option field

There are two additional environments: the Editor and the Help Facility. The Editor can be called from Command, Entry, or Option mode to edit the text currently being typed with the Edit command ☑. The Help Facility can only be called with the Help command (?) from Command mode.

In addition to the active environments, there are times when the cursor is temporarily inactive during program actions that require time to perform. A status message usually indicates such an action, which can only be interrupted by pressing the Break key.

Command Mode

Command mode allows single-key commands and extended commands to be typed any time the user is not in another mode. The user is automatically placed in Command mode if the first character typed is one of the following commands:

☑	Edit command
?	Help command
/	a Slash command
=	Select command
:	Goto command
;	Switch command (when there are two displayed windows)
›	Dive command (from the Variable, List, User Function, Plot and Table sheets, and Variable and User Function subsheets)

〈 Return command
(from Variable, List, and User Function subsheets)

! Action command
(from the Variable, Rule, Plot, and Table sheets; from List and User Function subsheets)

Certain commands require more than one keystroke: the Select command (=), the Goto command (:), the Slash commands (/), and the Help command (?). Of these, the Select command and some of the Slash commands prompt for a command option.

The Goto command (:), the Help Facility (?), the Storage command (/S), and the List command (/L) prompt for user responses on the Prompt/Error line. Commands that require typing on the Prompt/Error line in response to a prompt end in a colon. The cue appears on the Prompt/Error line and any characters typed must be entered with the Enter key. The Editor can be called on the Prompt/Error line by typing the Edit command ◪ any time before the Enter key is pressed. Valid user responses are prescribed by the particular command, or the computer file system in the case of file specifications.

Entry Mode

Entry mode allows characters to be typed and entered into an entry field. The user is automatically placed in Entry mode any time the cursor is positioned on an entry field and any character other than a command is typed. In Entry mode, the cue appears and all characters typed are inserted to its left; pressing the Backspace key deletes one character to the left of the cue.

Any characters typed into an entry field are accepted until the Enter key is pressed, at which time the item is checked for validity and entered. A valid item is determined by the type of entry field. If the item is determined to be invalid when the Enter key is pressed, the user is automatically placed in the Editor. The Break key can be used any time before the Enter key is pressed to cancel an uncompleted item. If the field contained a previous item, it is restored when the Break key is pressed.

Any characters typed after the Enter key or Break key is pressed, other than a command character, begin a new entry in the current field.

To enter a command character or a space as the first character of an entry field, call the Editor and use the special character as a regular entry character.

Option Mode

Option mode allows choices that perform certain actions to be typed into option fields. Options are entered without using the Enter key. The user is placed in Option mode any time the cursor is positioned on an option field and a character that is not a command character is typed. When a valid option is typed, the option is displayed in the field as appropriate and any immediate action taken. If an invalid option is typed, an error message on the Prompt/Error line indicates the valid option choices for that field.

The Editor

The Editor allows modification of the characters in a field or on the Prompt/Error line. The user is automatically placed in the Editor any time an entry error is made on an Entry field. The Editor can also be called from Command mode with the Edit Field command (/E), or from Entry mode or on the Prompt/Error line with the Edit command ◪. (See the Command section for further information on the Edit commands.)

The word "(Edit)" appears on the Prompt/Error line any time the Editor is being used.

The Editor is exited by pressing either the Enter key, to check the item for validity and complete it, or the Break key, to cancel the current unentered item and restore any previous item to the field, if one exists.

The Help Facility

The Help Facility provides on-line reminders about program features. It is called by typing the Help command (?). The Help topic prompt appears on the Prompt/Error line. There are three possible responses to this prompt any time it appears:

- Typing a second ? to get more information on using the Help Facility.
- Typing a topic name followed by the Enter key to locate all text on the specified topic.
- Pressing the Enter key to exit the Help Facility.

A list of valid topics can be displayed by typing the topic name "TOPICS" followed by the Enter key.

A single topic name may be associated with a number of blocks of text. If the text takes more than one page, a prompt appears for the next page. Typing **Y** continues to the next page; **N** returns immediately to the Help topic prompt. When the final text block has been displayed for a topic, the Help topic prompt re-appears. A prompt must appear, indicating the end of a page or that the last section of text has been displayed, before typing further.

A topic name can be corrected on the Prompt/Error line by using the Edit command ◪ any time before the Enter key is pressed. The Editor is exited and the search for the topic begun by pressing the Enter key.

If the specified topic is not available, a prompt appears offering more help. Typing **Y** displays the Help topic prompt again; typing **N** exits the Help Facility.

Help can be exited any time the Help topic prompt is displayed by pressing the Enter key, or interrupted at any point by pressing the Break key. The previous display and cursor position are returned.

Field Types

There are three types of fields in the TK!Solver program: entry fields, option fields, and output fields.

Entry Fields

Entry fields accept items typed in and entered by the user. The user is automatically placed in Entry mode whenever the cursor is positioned in an entry field and any character other than a command is typed. Text typed into a field is verified and completed by pressing the Enter key. If an item is not valid for the current field, pressing the Enter key produces an explanatory error message and places the user in the Editor.

Some entries automatically insert items in other fields. When the Variable Insert ON field on the Global Sheet is set to **Yes**, any previously undefined variable name entered in a Rule field is automatically inserted at the bottom of the Name column on the Variable Sheet. If a previously undefined list name is entered in any list name entry field, it is automatically inserted at the bottom of the Name column on the List Sheet. (Automatic list name insertion is independent of the Automatic Insert ON field on the Global Sheet.)

Once automatically inserted, a variable name or list name remains, even if the field in which it was entered is blanked or edited.

The maximum allowable length for a name, unit, comment, rule, filename, or setup string entry is 200 characters, although the display size is limited to the size available in a field or on the Status line. The maximum length for a variable name, symbolic value, or expression entered into a value, numeric value, or integer value field is also 200 characters. Text entries are case sensitive, meaning that distinction is made between uppercase and lowercase on computers that use both cases.

Numbers entered in value and numeric value entry fields have a maximum of 12 significant digits. The numeric range for values is system dependent. See the Hardware Reference section for the range for your system. Although the internal representation of number entries used in calculations is constant, numbers are displayed differently to fit in different display sizes: the Field item on the Status line is displayed up to the full 11 or 12 significant

digits; value and numeric value fields are displayed up to 10 significant digits. Numbers are rounded off or expressed in scientific notation upon entry to fit smaller display sizes. In greater display sizes, zeros are used to indicate the number of entered digits beyond the significant digits.

Integer entry limits are determined by the computer being used and the valid values for the particular field. (See the Hardware Reference section and the Sheet section for details.)

Value, numeric value, and integer entries can be entered as numbers or as expressions, if all variables have defined values. (See the Mathematics section for more information about valid expressions.)

Integer and numeric value entry fields contain default values provided by the program which can be changed but cannot be blanked.

Entry fields accept nine kinds of entries:

Name Entry
A name field is used to assign a name to a single value, a list of values, or a user-defined function. A relationship can be created between a list and a variable. A function can be formed by entering two list names in appropriate fields.

A name entry cannot begin with a number but thereafter can contain any combination of letters, digits, and the following characters up to 200 characters:

@ # $ % _

Spaces are not allowed.

A variable name can be entered in the Name field on the Variable Sheet; a user function name can be entered in the Name field on the User Function Sheet; all other name entry fields take list names.

Unit Name Entry
A unit name field is used to assign a unit of measurement name to a variable or list, or to define the conversion relationship between two unit names.

The conversion relationship between two unit names is assigned on the Unit Sheet. One unit name is assigned to each unit field on

the Unit Sheet, and each row defines one unit conversion relationship.

A pair of unit names can be assigned for each variable and each list. Variables have Calculation units and Display units; lists values have Storage units and Display units. Subsheets display the pair of unit names separately. The Unit fields on the Variable and List sheets show only a Display unit name.

The first unit name entered in a Unit field on the Variable Sheet assigns both the Calculation and the first Display unit for the corresponding variable; the first unit name entered in the Unit field on the List Sheet assigns both the Storage unit and the first Display unit for the corresponding list. Any subsequent unit name entry in the same field changes the Display unit only. (See the Unit Sheet in the Sheet section and Units in the Mathematics section for more information about unit conversions.)

A unit name entry can contain any combination of displaying characters, including internal spaces, up to 200 characters. Leading and trailing spaces are deleted upon entry.

Comment Entry
A comment field is used to enter a comment or to display text. Comments are not used by the program in calculations.

A comment entry can contain any printable characters and spaces up to 200 characters. The first character of a comment cannot be a command character or space unless it is edited in.

Comment entries on the Plot and Table sheets are used as display text.

Value Entry
A value field is used to assign a numeric or symbolic value to a variable, a list element, or a user function list element.

A value entry may be entered as a real number, a symbolic value, or any valid expression with defined values. The absolute value range for a real number entry is system dependent. See the Hardware Reference section for the range for your system. A symbolic value can be entered like a name entry, preceded by an apostrophe.

Numeric Value Entry
A numeric value field is used to assign parameters for the Solvers.

A numeric value entry may be entered as a real number or any valid expression with defined values. A numeric value field is assigned a positive real number value within the range for your system. See the Hardware Reference section for the range for your system. If a negative number is entered, its absolute value is used.

Numeric value entries are entered in the Comparison Tolerance and Typical Value fields on the Global Sheet. They are required entry fields with default values that can be changed but cannot be blanked.

Integer Entry
An integer field is used to assign parameters for program features. All integer entries are required and have default values.

An integer entry must be entered as a positive integer or any valid expression with defined values that evaluates to a positive integer. The maximum valid integer in a field is determined by the computer being used, but all integer fields have practical limits dependent upon their use. (See the Hardware Reference section or the Sheet section for more information.)

Rule Entry
A rule field is used to assign a statement relating variables, expressed as an equation between two expressions.

A rule entry must contain an equal sign and must conform to the conventions for valid rules outlined in the Mathematics section. Any characters following a quotation mark (") in a Rule field are interpreted as a comment. A rule field may contain up to 200 characters.

A rule entry can only be entered in the Rule fields on the Rule Sheet.

Filename Entry
A filename entry is used to specify the output device name or the file specification for a file used by the program.

A filename entry can be entered up to 200 characters, but the length and validity of a file specification is usually determined by the operating system of the computer. (See the Hardware Reference section for more information.)

A filename entry can be entered on the Global Sheet in the Printer Device or Filename field, or on the Prompt/Error line in response to a prompt for a filename in the Storage command.

Setup String Entry
A setup string field is used to enter a string that specifies printing options for the attached printer.

A setup string entry can include displaying characters which represent control and escape sequences that direct the printer to print titles, alter type size, underscore or boldface, issue form feeds, etc. The first character cannot be a command character unless edited in. A setup string entry field can accept up to 200 characters, but valid setup strings are dependent on the computer and printer being used. (See the Global Sheet in the Sheets section, the Hardware Reference section, and the manuals that accompany your computer and printer for more information.)

A setup string entry can only be entered in the Printer Setup String field on the Global Sheet.

Option Fields

Option fields accept a single character, without the Enter key, selected by the user from a limited set of possibilities and interpreted by the program. If an invalid option is typed, an error message displays listing the valid options.

There are two kinds of option fields: those in which the selected character is used to perform a task (such as the **I** or **L** options in the Status fields of the Variable Sheet); and those in which the character is expanded into a full definition (such as the **T** option in the Mapping fields of the User Function Sheet).

Some option fields have default values that can be changed but they cannot be blanked. Field names ending in "ON" require **Y** or **N** options. Field names containing "or" offer a choice between two requirements.

The Status fields on the Variable Sheet are special option fields that are also used by the program to display error codes. The Character fields on the Plot Sheet may contain any displaying character or be blanked to display nothing.

Output Fields

Output fields are used by the program to display information. The cursor can be moved to any output field except Element fields on List and User Function subsheets, but the user cannot type in output fields. The Status fields on the Variable Sheet are both option and output fields. The Value field on List subsheets and the Domain and Range fields on User Function subsheets are both Value Entry and Output fields.

On List and User Function subsheets, element numbers are automatically assigned in a column to the left of values as they are entered.

Error Conditions

Error messages for the TK!Solver program appear on the Prompt/Error line at the top of the screen, or, in the case of solution errors, in the Status columns of the Variable or Rule sheets.

Error messages are sometimes provided on the bottom of the screen for hardware problems, such as a missing disk or printer. These errors usually include an instruction about what to do.

See the Appendix section for a list of each error message provided by the program and the appropriate actions to take.

The Break key can be used to cancel an extended command, discard a partially entered item, or interrupt an action. The computer produces an error signal and cancels the activity.

Entry Errors

When an attempt is made to enter an invalid item in an entry field, the computer produces an error signal, an error message appears on the Prompt/Error line, the user is placed in the Editor, and the cue appears to the right of the first recognized error in the field. The error message disappears when the Break key is pressed to cancel the entry, or when the Enter key is pressed to enter a valid item.

Option Errors

When an attempt is made to type an invalid option in an option field, the computer produces an error signal, and an error message appears on the Prompt/Error line providing the valid options. The error message disappears with the next valid keystroke.

Command Errors

When an attempt is made to type invalid command keystrokes, or a command at an invalid time, the computer produces an error signal, and the command prompt, if any, disappears from the Prompt/Error line.

Unit Errors

When an attempt is made to convert the value of a variable to another unit with no unit conversion defined, a question mark (?) appears to the left of the value in the corresponding value field.

If units are used for a variable or a list, both the Calculation unit, or Storage unit, and the Display unit must be defined. An error is indicated if one of the two units has been independently blanked on a subsheet: a question mark appears to the left of any unit name that has no defined Calculation or Storage unit; a question mark appears to the right of any unit name with no defined Display unit.

The question mark in a value field or a unit field disappears when a proper unit conversion definition is provided or all units are blanked for the variable values in question.

Solution Errors

An unsolved model is indicated by the Solution Indicator in the upper righthand corner of the screen and certain symbols in the Status fields of the Variable and Rule sheets.

The Solution Indicator (!) appears whenever a change is made that may affect the solution of a model, such as entering a new value or editing a rule. It disappears when any equation is solved and there are no errors in the model.

Before a solution, an asterisk (*) appears in the Status column of the Rule Sheet to the left of all unsatisfied rules. An unsatisfied rule is one that has not yet been evaluated successfully by the solvers.

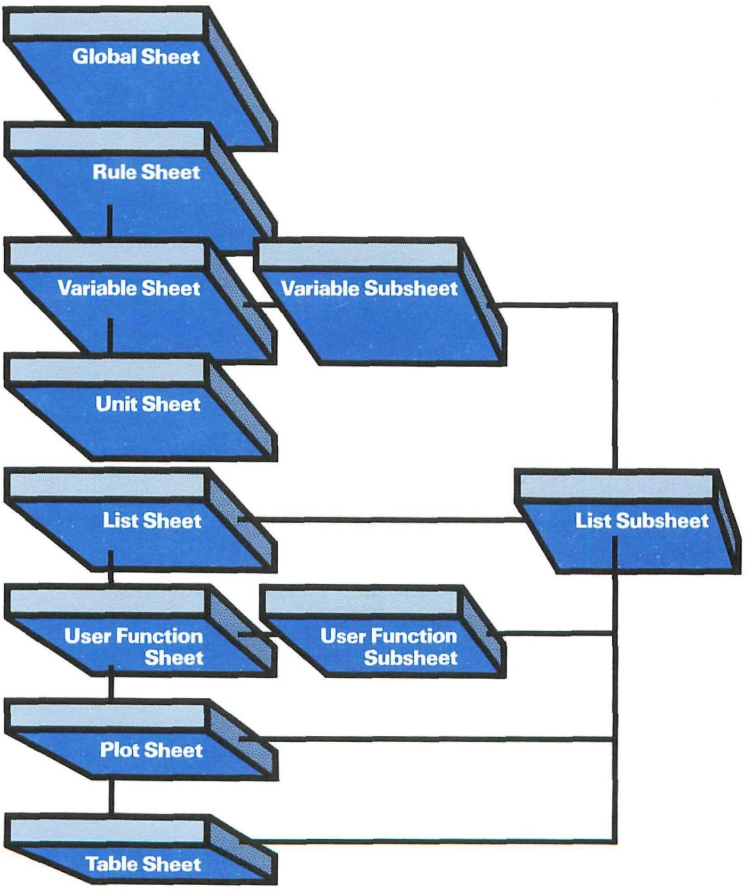
After a solution has been attempted and an error found, the greater than symbol (>) appears in the Status column of the Rule Sheet identifying equations in error. The program recognizes an equation containing a calculation error (such as division by zero) and sets of equations that are inconsistent with each other. In a set of conflicting equations, usually the first equation in conflict with a previously evaluated rule is marked. Whenever the TK!Solver program encounters inconsistent or contradictory data in an equation, it stops solving.

When solution fails due to an error and rules are marked, the greater than symbol (›) also appears in the Status column of the Variable Sheet to indicate the variables involved in the failure. Variables are marked for overdefinition or being involved in a calculation error.

An explanatory message is provided for each Status field symbol indicating the type of failure. It can be viewed by moving the cursor to the Status field in question and looking at the explanation of the symbol on the Status line.

Error symbols in Status fields are displayed until the error condition no longer exists. Asterisks remain until a rule is satisfied or determined to contain an error by one of the solvers.

Sheets


Global Sheet
Rule Sheet
Variable Sheet
Variable Subsheet
Unit Sheet
List Sheet
List Subsheet
User Function Sheet
User Function Subsheet
Plot Sheet
Table Sheet

Sheets and Subsheets

There are eight sheets and three subsheets in the TK!Solver program.

This section describes the sheets and subsheets available with the TK!Solver program in the following order:

Variable Sheet
Variable Subsheets
Rule Sheet
Unit Sheet
Global Sheet
List Sheet
List Subsheets
User Function Sheet
User Function Subsheets
Plot Sheet
Table Sheet

The description of each sheet or type of subsheet contains:

- its purpose
- its picture, containing data from the mortgage model used in the Instruction Manual
- how to access it and the subsheets associated with it
- its initial cursor position
- its fields

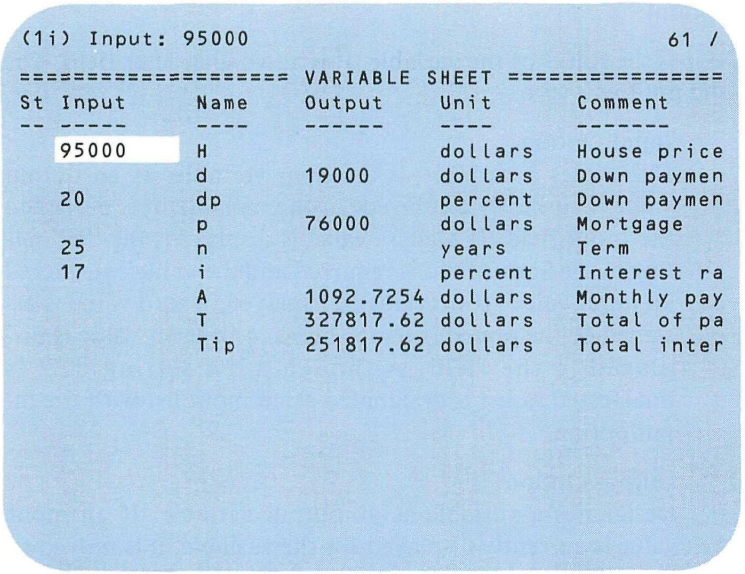
Each field listing contains:

- its label, followed by its label letter in parentheses (label abbreviations on the screen are also spelled out here)
- its field type, in brackets
- any default value for the field
- a description of the field
- a description of any field options

See the Overview section for more information about terminology used to describe sheets and the conventions for valid field types.

The Variable Sheet

The Variable Sheet contains data describing the variables of a model. Each row can contain the description for one variable. There is a Variable subsheet displaying more information about each variable; items entered into fields on the Variable Sheet are updated simultaneously on corresponding fields on Variable subsheets.



```
(1i) Input: 95000                                          61 /

==================== VARIABLE SHEET ================
St Input      Name     Output      Unit       Comment
-- -----      ----     ------      ----       -------
   95000      H                    dollars    House price
              d        19000       dollars    Down paymen
   20         dp                   percent    Down paymen
              p        76000       dollars    Mortgage
   25         n                    years      Term
   17         i                    percent    Interest ra
              A        1092.7254   dollars    Monthly pay
              T        327817.62   dollars    Total of pa
              Tip      251817.62   dollars    Total inter
```

Access:
The Variable Sheet appears in the top window at program startup. It can be displayed in the current window by typing **=V**, or in a new window by typing **/WV** from a one-window display.

A Variable subsheet for a variable can be reached by typing the Dive command (**)**) with the cursor on any field on the variable row.

The List subsheet for a list associated with a variable can be reached by placing the cursor on any field in the variable row and typing the Dive command (**)**) twice, once into the Variable subsheet, and a second time into the List subsheet for the associated list. The cursor may be positioned on any field on the Variable subsheet.

Initial Cursor Position:
The Input field on the first row

Fields:
Each variable is described by a row containing six fields, located in columns under the labels in the Label line at the top of the sheet. The Variable Sheet fields are:

St [Status] (s)
[Option]

Defines the status of the variable. It is a two-character field. The valid options are:

I Input Option
Designates a variable as an input variable. If an output value is currently displayed for the variable, it is moved to the Input field. If a guess value is displayed, the "G" setting in the Status field is removed and the value considered an input value. If no value is displayed, a zero value is assigned or the previous input, guess, or output value is displayed in the field. Although a list setting "L" is unaffected, a list is designated as an input list with the input option.

O Output Option
Designates a variable as an output variable. If an input value is currently displayed for the variable, it is moved to the Output field. If a guess value is displayed, the "G" setting in the Status field is removed and the value moved to the Output field. If no value is displayed, a zero value is assigned or the previously input or output value is displayed in the field. Although a list setting "L" is unaffected, a list is designated as an output list with the output option.

L List Option
Associates a variable with a list name. An "L" is displayed in the first character position of the Status field. The variable name is assigned by default to the Associated List field on the Variable subsheet for that variable and a list by that name is created on the List Sheet if it does not already exist. The default list name may be changed in the Associated List field on the Variable subsheets. Lists are inserted at the

bottom of the List Sheet in the order they are associated and created. If there are Calculation and Display units for a variable when the list association is made, they are assigned as the Storage and Display units for the list. (If units are assigned to a variable after the list association is already made, they are not carried over.) Typing **L** a second time removes the "L" from the Status field and disassociates the list, leaving it listed on the List Sheet but no longer connected with the variable.

G Guess Option

Designates a variable value as a starting guess value for a variable for use with the Iterative Solver. A "G" is displayed in the second character position of the Status field and the word "Guess" displayed on the Status line. If an output value is currently displayed for the variable, it is moved to the Input field. If an input value is displayed, it is considered a guess value. If no value is displayed for the variable, a zero value is assigned or the previous blanked input, guess, or output value is displayed in the field. The previous input value may be changed. Typing **G** a second time removes the "G" from the Status field and leaves the value in the Input field as an input value. A list setting "L" is not changed. If a list is associated, a "G" in the Status field designates the elements in the list as guess values. The results of the Iterative solution are written into the list for each element.

B Blank Option

Blanks any values for a variable from the Input or Output fields and any "G" option from the Status field. A list setting "L" is not changed. Blanked values can be recovered by typing **I**, **O**, or **G** in the Status field. Blanked values are saved when a model is saved.

[Output]

If solution fails, the program displays an error symbol (⟩) in the second character position of the Status field of each variable involved in an error condition. If the cursor is moved to the Status field containing the error symbol, its explanatory message is displayed on the Status line at the top of the screen.

Input (i)
[Value Entry]

Accepts an input value for a variable. An input value can be any valid numeric value or expression, or a symbolic value. Entering an input value erases any value from the Output field for that variable entity.

If Calculation and Display units are assigned to the Unit field for the variable, changing the Display unit immediately converts the input value to a value in the new Display unit. If the conversion between the Calculation unit and Display unit is undefined, the value is displayed in the Calculation unit, preceded by a question mark (?).

If a numeric value has been blanked, it can be recovered by typing **I** or **G** in the Status field.

See the Overview and Mathematics sections for more information about valid value entries.

Name (n)
[Name Entry]

Accepts a unique variable name. A variable must be named if it is to be associated with a list.

If the Variable Insert ON field on the Global Sheet is set to "Yes", any variable names entered in a Rule field on the Rule Sheet that do not previously exist are automatically inserted at the bottom of the Name column. Once inserted on the Variable Sheet a variable remains even if the rule in which it was entered is deleted or edited.

Output (o)
[Output]

Displays the output value for a variable if a solution produces a value or if a value is moved there by typing **O** in the Status field.

If different Calculation and Display units are assigned to the variable, the output value appears in the Display unit. If the conversion between the designated Calculation unit and Display unit is undefined, the value is displayed in the Calculation unit, preceded by a question mark (?).

If the value has been blanked, it can be recovered by typing **O** in the Status field.

Unit (u)
[Unit Name Entry]

Assigns the Calculation and Display unit names for the variable. If the Unit field is blank, the first unit entered becomes both the Calculation and Display units for the variable. Any subsequent entry changes only the Display unit. The Blank command (/B) can be used on the Variable Sheet to blank both the Calculation and Display units for a variable simultaneously.

The unit conversion must be defined explicitly on the Unit Sheet, or the two units must be defined in terms of a common third unit. If the conversion is defined, the numeric value in the Input or Output field (and Input Value or Output value, and First Guess fields on a Variable subsheet) is immediately converted to its value in the Display unit whenever the Display unit name is changed.

If the conversion between the Calculation unit and the Display unit is not defined on the Unit Sheet, any value in the Input or Output field is expressed in the Calculation unit and preceded by question mark (?).

Any variable associated with units must have both a Calculation and Display unit assigned at all times, although they can be identical. If the Calculation unit name has been blanked from the Variable subsheet without also blanking the Display unit, the unit name appears preceded by a question mark. If the Display unit has been independently blanked, the unit name appears followed by a question mark.

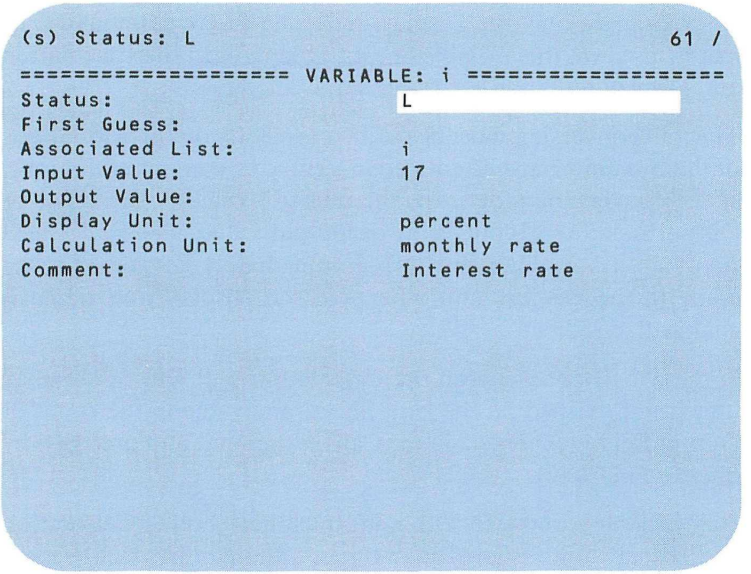
For more information about unit conversion, see the Mathematics section.

Comment (c)
[Comment Entry]

Accepts a comment or description of the variable. Any characters entered into this field are ignored by the program during solving. Comments can be especially useful if a model is to be saved for later use.

Variable Subsheets

Each Variable subsheet contains a detailed description of one variable defined on the Variable Sheet. There is one Variable subsheet for each variable, whether named or not, on the Variable Sheet. Items entered in fields on a Variable subsheet are updated simultaneously on corresponding fields on the Variable Sheet.



Access:
A Variable subsheet is accessed by placing the cursor anywhere on the row on the Variable Sheet for the variable to be examined and typing the Dive command (›). Typing the Return command (‹) returns to the Variable Sheet.

If a list name has been assigned to the Associated List field, the associated List subsheet can be reached by diving again with the cursor positioned on any field on the Variable subsheet.

Initial Cursor Position:
The Status field

Fields:
The variable defined on a Variable subsheet is described by eight header fields, with a label and a field on each line. Fields appearing on both a Variable subsheet and the Variable Sheet behave in the same way unless otherwise noted. The Variable subsheet fields are:

Status: (s)
[Option/Output]

Same as the Status field on the Variable Sheet, except that the "G" option is spelled out as "Guess".

First Guess: (f)
[Value Entry]

Provides a starting guess value for automatic use of the Iterative Solver. The first guess value is used if the Automatic Iteration ON field on the Global Sheet is set to "Yes" and the Direct Solver fails to find a value for the variable.

A first guess value can be any valid numeric value or expression. If Calculation and Display units are assigned for the variable, changing the Display unit name immediately converts the guess value to the new Display unit. Automatic iteration is most useful if the problem values are within a narrow range.

Associated List: (a)
[Name Entry]

Associates the variable with a list for use by the List Solver. When **L** is typed in the Status field on the Variable Sheet or a Variable subsheet, the variable name is assigned as the list name by default. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet. The default list name can be changed to any other list name by typing the new name into the Associated List field. Entering a list name in the Associated List field automatically generates an "L" in the Status field for both the Variable Sheet and subsheet.

An association may be cleared by blanking the Associated List field or by typing a second "L" in the Status field on either the Variable subsheet or the Variable Sheet. Once inserted on the List Sheet, a list remains even if the list association is removed.

Input Value: (i)
[Value Entry]

Same as the Input field on the Variable Sheet except that the value is displayed with up to 12 significant digits.

Output Value: (o)
[Output]

Same as the Output field on the Variable Sheet except that the value is displayed with up to 12 significant digits.

Display Unit: (d)
[Unit Name Entry]

Same as the latest unit name entry in the Unit field on the Variable Sheet. Displays the current Display unit for the variable. The Display unit can be assigned on the Variable Sheet by changing the unit name in the Unit field, or on the Variable subsheet by typing directly into the the Display Unit field. The Display unit on a Variable subsheet should not be blanked without also blanking the Calculation unit. Each time the Display unit is changed and the conversion is defined on the Unit Sheet, the corresponding input and output and first guess values are converted.

Calculation Unit: (u)
[Unit Name Entry]

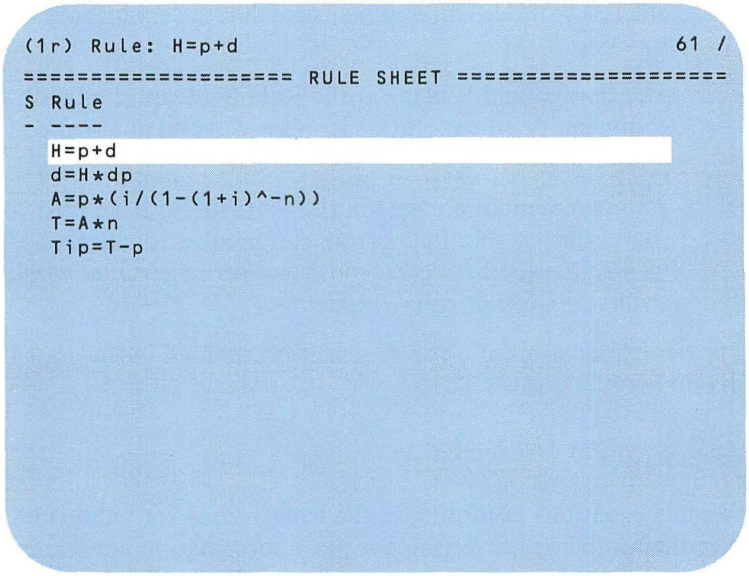
Same as the first unit name entry in the Unit field on the Variable Sheet. Displays the current Calculation unit for the variable. The Calculation unit can be assigned on the Variable Sheet as the first unit name entered in the Unit field, or on the Variable subsheet by typing directly into the the Calculation Unit field. The Calculation unit on a Variable subsheet should not be blanked without also blanking the Display unit.

Comment: (c)
[Comment Entry]

Same as the comment field on the Variable Sheet. It is not used in solutions.

The Rule Sheet

The Rule Sheet contains the equations for a model. Each row can contain one equation and its status, as well as a comment.



Access:
The Rule Sheet appears in the bottom window at program startup. It can be displayed in the current window by typing **=R**, or in a new window by typing **/WR** from a one-window display.

Initial Cursor Position:
The Rule field on the first row

Fields:
Each rule is described by a row containing two fields located in columns under the labels in the Label line at the top of the sheet. The Rule Sheet fields are:

S [Status] (s)
[Output]

Used by the program to provide a message about the status of the rule in the corresponding row. An explanation of the character

displayed in a Status field can be viewed on the Status line at the top of the screen by moving the cursor to the Status field in question.

* An asterisk appears in the Status field whenever a rule is entered or modified, and indicates that the equation is unsatisfied.

The Status field is blank if the Rule field contains only a comment, or when solution is successful and no error conditions arise.

› An error symbol appears in the Status field if an error is found during solution. Errors can involve function arguments, division by zero, inconsistent variable values within a rule, or contradictory rules.

Further explanation of error messages is provided in the appendix on Error Messages.

Rule (r)
[Rule Entry]

Accepts equations according to the conventions for valid rules. (See the Mathematics section for more information concerning valid rules.)

A rule field can contain an equation, a comment, or both. All text after a quotation mark (") in a Rule field is regarded as a comment.

A quotation mark in front of a rule excludes a rule from being evaluated by the solvers.

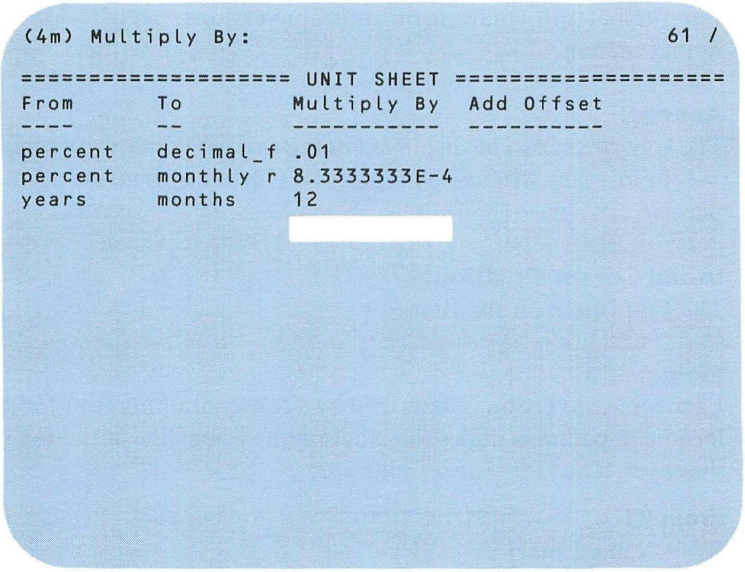
If the Variable Insert ON field on the Global Sheet is set to "Yes" (the default setting), any variables not already on the Variable Sheet are automatically inserted at the bottom of the Name column on the Variable Sheet when a rule is entered. Once assigned, the variable names remain even if they are removed from the equations on the Rule Sheet.

Error messages appear on the Prompt/Error line for any invalid rule entries. User functions must be defined on the User Function Sheet before they can be used in rules.

See the Mathematics section for more information concerning rule evaluation.

The Unit Sheet

The Unit Sheet contains definitions for conversions between units of measurement used in a model. Each row on the Unit Sheet describes a conversion between two units of measure.



A unit conversion must be defined on the Unit Sheet before it can be used on the Variable or List Sheets or subsheets. Once a conversion is defined, it can be used in either direction. Two units can be used in conversions even though their conversion is not explicitly defined on the Unit Sheet, if they are each defined in terms of a third common unit.

The management of units is left to the user. It is recommended that all variables on the Variable Sheet be assigned unit names at the same time, prior to value assignment. Since rule evaluation is performed in Calculation units, it is important to assign Calculation units for variables so that the results of solutions are consistent among related variables. Check the Display unit, and change it if necessary before you enter any input value. It is necessary to assign units to a variable before associating that variable with a list, if unit conversions are being used. The Storage unit of a list

should have a conversion defined on the Unit Sheet with the Calculation unit of its related variable or else be identical. If a function argument value is assigned to a variable, the Display units for the domain and range lists must be the same as the Calculation unit for the variable.

For further information on unit conversions, see the Mathematics section.

Access:
The Unit Sheet can be displayed in the current window by typing =**U** or in a new window by typing /**WU** from a one-window display.

Initial Cursor Position:
The From field on the first row

Fields:
Each unit conversion is described by a row containing four fields, located in columns under the labels in the Label line at the top of the sheet. The Unit Sheet fields are:

From (f)
[Unit Name Entry]

Names the first unit. A single value of the first unit equals the specified multiple of the second unit.

See the Overview section for valid unit name entries.

To (t)
[Unit Name Entry]

Names the second unit. The second unit is defined in terms of a single first unit.

Multiply By (m)
[Value Entry]

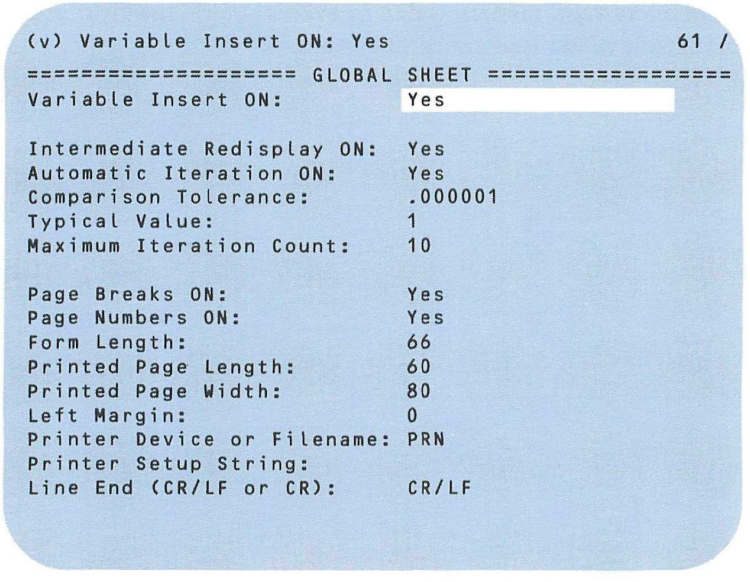
Defines the number of second units necessary to equal a single first unit. A Multiply By factor entry can be any valid numeric value or expression. If the field is left blank, one second unit equals one first unit.

Add Offset (a)
[Value Entry]

Defines any difference between the zero points of the scales used for the two units. This offset value must be in terms of the second unit. For example, converting from degrees Centigrade to degrees Fahrenheit requires an offset of 32. The addition to align the scales is performed after the multiplication. An offset item can be any valid numeric value or expression. If the field is left blank, the offset is assumed to be zero.

The Global Sheet

The Global Sheet contains global settings and program defaults. Each row contains one global setting labeled on the left. Different computer systems have different default values; for more information about the global settings for your system, see the Hardware Reference section.



Access:
The Global Sheet can be displayed in the current window by typing =**G** or in a new window by typing /**WG** from a one-window display.

Initial Cursor Position:
The Variable Insert ON field

Fields:
The global settings for the program are described by fifteen header fields, with a label and a field on each line. The Global Sheet is organized in three sections: the first section contains one field affecting the entry of variables; the second section contains five fields affecting the use of the solvers; the third section contains nine fields affecting use of the printer. The printer options

are used by the Print command (/P) and the Plot and Table sheets when their Printer options are selected. The Global Sheet fields are:

Variable Insert ON: (v)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Controls automatic insertion of the variables used in rules onto the Variable Sheet. If this field is set to "Yes", the program automatically enters variable names not already on the Variable Sheet at the bottom of the Name column on the Variable Sheet when an equation is entered on the Rule Sheet. Once inserted, variable names remain on the Variable Sheet even if they are edited or blanked from the Rule Sheet. If this field is set to "No", variable names must be inserted by hand on the Variable Sheet.

Intermediate Redisplay ON: (r)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Controls the display of intermediate values during Iterative and list solution. If the field is set to "Yes", the results for intermediate values are displayed on the Variable Sheet during solution. If the field is set to "No", the screen retains the initial values until the end of the solution and is updated at the end of Iterative solution only. The status message with the element number or the iteration step is still displayed on the Prompt/Error line. Solution is faster when the field is set to "No".

Automatic Iteration ON: (a)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Allows automatic iteration to be used when a Direct solution fails. If the field is set to "Yes" and first guess values are assigned on Variable subsheets, the Iterative Solver is called automatically

after the Direct Solver fails to reach a solution. If the field is set to "No" or if first guess values have not been entered, solution is terminated when the Direct Solver fails.

Comparison Tolerance: (c)
[Required Numeric Value Entry]
Default: .000001

Defines equality for variables and expressions during solving. Two values are considered equal if the absolute value of their difference divided by the larger of the two values is less than the comparison tolerance. A comparison tolerance can be any positive valid numeric value or expression.

See the Mathematics section for more information.

Typical Value: (t)
[Required Numeric Value Entry]
Default: 1

Used in conjunction with the comparison tolerance to determine at what point values are considered near zero. The product of the typical value and the comparison tolerance determines this point. Whenever values are equal to or less than this product, the denominator of the comparison tolerance formula is changed to the typical value. As the typical value increases in size for a constant comparison tolerance, there is a concurrent decrease in the precision of equality determination. A typical value can be any positive valid numeric value or expression.

See the Mathematics section for more information.

Maximum Iteration Count: (i)
[Required Integer Entry]
Default: 10

Defines the maximum number of iterations to be tried by the Iterative Solver before it stops. Any integer between 1 and 10000 may be assigned, although values higher than 20 are not recommended.

See the Mathematics section for more information.

Page Breaks ON: (b)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Defines page breaks for the Print command and display of tables. If the field is set to "Yes", the output from the Print command (/P) or the Table Sheet (!) is printed with breaks after each printed page length. Table displays repeat any specified title and headers on each page.

If the field is set to "No", output is generated as a single continuous sheet. When page breaks are off: page numbers and form length settings are ignored; titles and headers for vertical tables appear at the top of the listing; titles for horizontal tables appear in between each segment of lists, and headers appear on every line.

Page Numbers ON: (n)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Controls page numbering for the Print command, and plot and table displays. If the field is set to "Yes", the output from the Print command (/P) or the Plot and Table Sheets (!) are printed with page numbers on the bottom of each page, preceded by the word "Page". When page numbers are used, the amount of printed material is decreased by two lines at the bottom of each page. If the field is set to "No", or page breaks are not in effect, page numbers are not used.

Form Length: (f)
[Required Integer Entry]
Default: 66

Defines the total number of lines on each sheet of paper. Any integer between 6 and 1000 may be assigned. The amount of space left between pages is determined by the form length minus the printed page length. If the printed page length is greater than or equal to the form length, the printed page length value is used and no space is left between pages. On perforated paper, leave a margin at the top and bottom of the page by dividing in half the difference between printed page length and the form length and moving the printer head down that number of lines below the perforations.

When page breaks are off, the form length setting is ignored.

Printed Page Length: (l)
[Required Integer Entry]
Default: 60

Defines the number of lines available for printing on each page. Any integer between 6 and 1000 may be assigned. The setting for printed page length must be less than or equal to form length. If the printed page length is greater than the form length, the form length is ignored.

The amount of information printed per page is decreased from the printed page length setting by two lines each for page numbers and titles that were entered on the Table or Plot sheets.

The length of a printed plot can be increased or decreased by adjusting the printed page length, whether page breaks are on or not.

Printed Page Width: (w)
[Required Integer Entry]
Default: 80

Defines the number of character positions available to the program for printing across the width of the page. Any integer between 6 and 200 may be assigned. The page width and the left margin setting together should be less than or equal to the physical width of the paper being used or information on the right end of each line will be lost.

Left Margin: (m)
[Required Integer Entry]
Default: 0

Defines the number of character positions to indent before the first characters of the line are printed. Any integer between 0 and 200 may be assigned. The left margin setting plus the printed page width should be smaller than or equal to the width of the paper or information on the right end of each line will be lost.

Printer Device or Filename: (d)
[Filename Entry]
Default: [Computer-dependent]

Defines the device or filename to which output is printed. The device or filename is used for output from the Print command (/P) or the Plot Sheet and Table Sheets (!) when their "Screen or Printer" fields are set to "Printer".

Most operating systems have a reserved name used to refer to the attached printing device. (See the Hardware Reference section for the default option and the manual that accompanies your operating system for more information.) Any other file specification directs output to the named file. Valid file specifications are determined by the operating system. (See the Hardware Reference section or your operating system manual for more information.) Error messages are provided when an invalid file specification is used, not when it is entered. (See the appendix on Error Messages for more information about error messages.)

If the field is blanked, the output is displayed on the screen using the printer options set in the other printer definition fields. The Enter key must be pressed to display each new page and return to the previous display after the last page.

Printer Setup String: (s)
[Setup String Entry]

Defines a special string sent to direct the printer at the beginning of any printing action. A setup string is normally defined by a series of printable ASCII characters and control code sequences. They are represented in the TK!Solver program by special conventions:

^	Indicates the beginning of a control code sequence
^^	Sends the caret character itself (^)
^R	Sends a carriage return character
^L	Sends a line feed character
^C	Precedes a Control character
^E	Represents an Escape character
^H	Precedes a two-digit hexadecimal code for an ASCII character

For example, Control A is represented by "^CA", and "^CA" is equivalent to "^H01". Some printer settings remain in effect until explicitly turned off in a subsequent print action with a newly specified printer setup string.

See the Hardware Reference section for more information concerning setup strings.

Line End: (e)
[Required Option]
Default: [Computer-dependent]

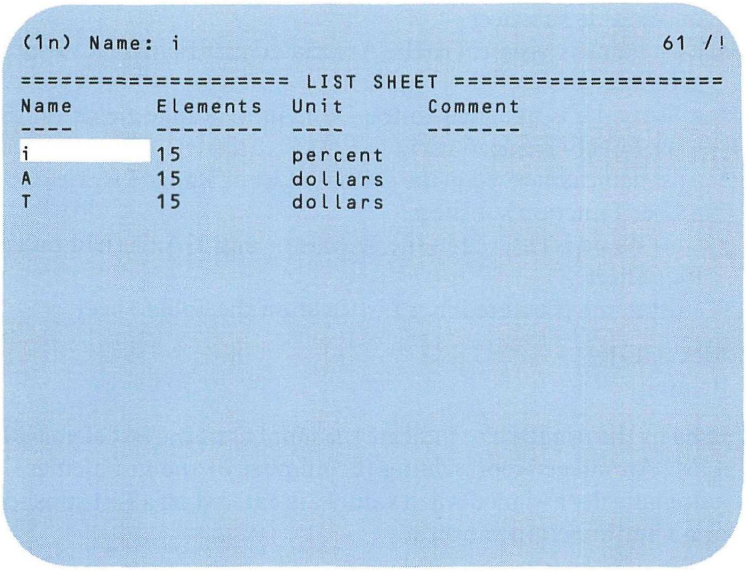
& Carriage Return & Line Feed
C Carriage Return
L Line Feed
S System-defined

Defines the line end sequence to be used for the printer. Systems that do not use carriage return or line feed have a default defined for that system. If the default is **&**, **C**, or **L**, the **S** option provides the correct code to the printer. If the default option is **S**, it cannot be changed.

This field is not changed by the Reset command (/R) and it is not saved or loaded as part of a model. See the Hardware Reference section for the default for your computer. See your hardware and printer manuals for more information about the sequence required by your printer.

The List Sheet

The List Sheet contains a table of all lists created in the program. A list is a set of values assigned to positional element numbers. Lists can have blank values. Each row can contain the description for one list. There is a List subsheet displaying more information about each list; items entered in fields on the List Sheet are updated simultaneously on corresponding fields on List subsheets.



Access:
The List Sheet can be displayed in the current window by typing =**L** or in a new window by typing /**WL** from a one-window display.

A List subsheet for a list can be reached by typing the Dive command (**)**) with the cursor on any field on the list row.

Initial Cursor Position:
The Name field on the first row

Fields:
Each list is described by a row containing four fields, located in columns under the labels at the top of the sheet. The List Sheet fields are:

Name (n)
[Name Entry]

Accepts a unique name for a list. A list must be named if it is to be associated with a variable or be considered a domain or range list.

A list name is automatically inserted at the bottom of the Name column, if it does not previously exist and:

- an **L** option is typed into the Status field on the Variable Sheet or Variable subsheet
- a list name is assigned to the Associated List field on a Variable subsheet
- a list name is assigned to the Domain or Range fields on the User Function Sheet
- a list name is entered in the Domain List or Range List fields on a User Function subsheet
- a list name is entered in the X-Axis or any Y-Axis field on the Plot Sheet
- a list name is entered in a List field on the Table Sheet

Elements (e)
[Output]

Used by the program to indicate the number of the last element in a list. An integer representing the number of the last element is automatically assigned when values are entered on a List subsheet or a User Function subsheet.

Unit (u)
[Unit Name Entry]

Assigns the Storage and Display unit names for the list. If the Unit field is blank, the first unit name entered becomes both the Storage and Display unit for the list. Any subsequent entry changes only the Display unit. The Blank command (/B) can be used on the List Sheet to blank both the Storage and Display units for a list simultaneously.

If Display and Storage units for a list are different, their unit conversion must be defined on the Unit Sheet, or the two units must be defined in terms of a common third unit. If the conversion is defined, the numeric values in the Value column on the List subsheet are immediately converted to their values in the Display unit whenever the Display unit name is changed.

If a list is associated with units, both a Storage and Display unit should be defined at all times, although they can be identical. If the Storage unit name has been blanked from the List subsheet without also blanking the Display unit, the unit name appears preceded by a question mark (?). If the Display unit has been independently blanked, the unit name appears followed by a question mark.

If a list is associated with a variable, its Storage unit must have a defined relationship to the Calculation unit of the variable, or the results may be unreliable.

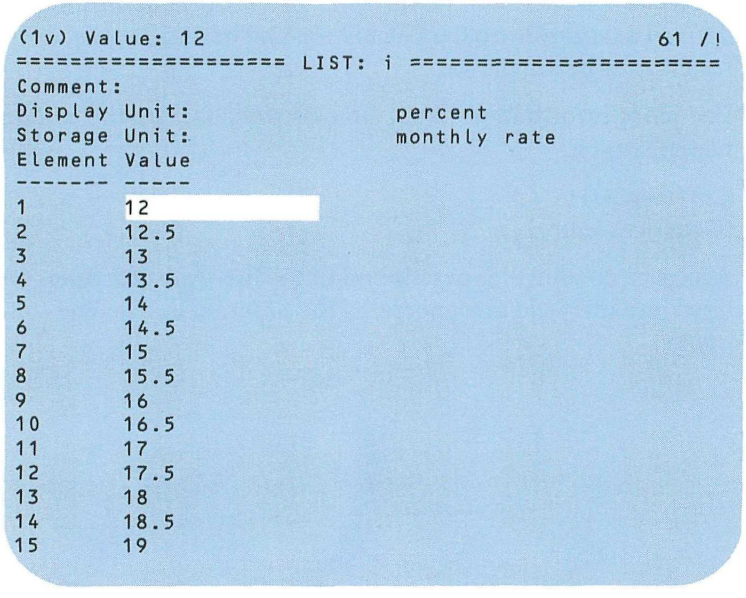
For more information about unit conversion, see the Mathematics section.

Comment (c)
[Comment Entry]

Accepts a comment or description of the list. Any characters entered into this field are ignored by the program in reaching solutions.

List Subsheets

Each List subsheet contains a detailed description of one list defined on the List Sheet. There is one List subsheet for each list on the List Sheet, whether named or not. Items entered in fields on a List subsheet are updated simultaneously on corresponding fields on the List Sheet.



A List subsheet is created for each of the two lists on a User Function subsheet. Values entered in the domain and range lists on a User Function subsheet are automatically entered in the Value fields of the two corresponding List subsheets. Changes made to either the List subsheets or User Function subsheet appear on all corresponding sheets and subsheets.

Access:
A particular List subsheet is accessed by placing the cursor anywhere on the row on the List Sheet for the list to be examined and typing the Dive command (›). Typing the Return command (‹) returns to the List Sheet.

A List subsheet can also be accessed by typing the Dive command (›) from:

- any field on a Variable subsheet with an associated list
- the Domain List, Range List, Domain, and Range fields of a User Function subsheet
- the X-Axis, Y-Axis, or Character fields on the Plot Sheet
- any row in the Field Area of the Table Sheet

Initial Cursor Position:
The first Value field

Fields:
The list defined on a List subsheet is described by three header fields at the top of the sheet, with a field label and a field on each line. Each list element is described by a row in the Field Area containing two fields in columns under labels. The values in a list are identified by an element number automatically generated by the program on the left side of the sheet as values are entered. Corresponding fields appearing on a List subsheet and a User Function subsheet or the List Sheet behave in the same way unless otherwise noted. The List subsheet fields are:

Comment: (c)
[Comment Entry]

Same as the Comment field on the List Sheet.

Display Unit: (d)
[Unit Name Entry]

Same as the latest entry in the Unit field on the List Sheet. Defines the unit the values of a list are currently displayed in. The Display unit can be assigned on the List Sheet by changing the previous unit name in the Unit field or on the List subsheet by typing directly into the the Display Unit field. The Display unit on a List subsheet should not be blanked without also blanking the Storage unit. Each time the Display unit is changed, the values in the Value column are automatically converted.

If the List subsheet is for the domain or range list for a user function, the functional relationship is determined strictly in the Display unit values.

Storage Unit: (s)
[Unit Name Entry]

Same as the first entry in the Unit field on the List Sheet. Defines the unit in which the values of a list are stored and the unit converted to Calculation units when associated with variables. The Storage unit can be assigned on the List Sheet as the first unit name entered in the Unit field, or on the List subsheet by typing directly into the the Storage Unit field. The Storage unit on a List subsheet should not be blanked without also blanking the Display unit.

Lists saved with the Save File (/SS) and DIF Save (/S#S) options of the Storage command are saved with both Storage and Display units.

Element
[Restricted Output]

Element numbers are automatically and sequentially assigned as values are entered. If a value is blanked, its element number is not displayed but the row remains reserved for it. If a value is deleted, its element number is reassigned to the next undeleted value. Some built-in functions and certain options of the List command use the element number to refer to a list value. The cursor cannot be moved to the Element column.

Value (v)
[Value Entry/Output]

Assigns a numeric or symbolic value to a list element. A value entry can be any valid TK!Solver expression.

If the conversion is not defined between the Display unit and Storage unit on the Unit Sheet, a question mark (?) precedes values expressed in the Storage unit in the Value column.

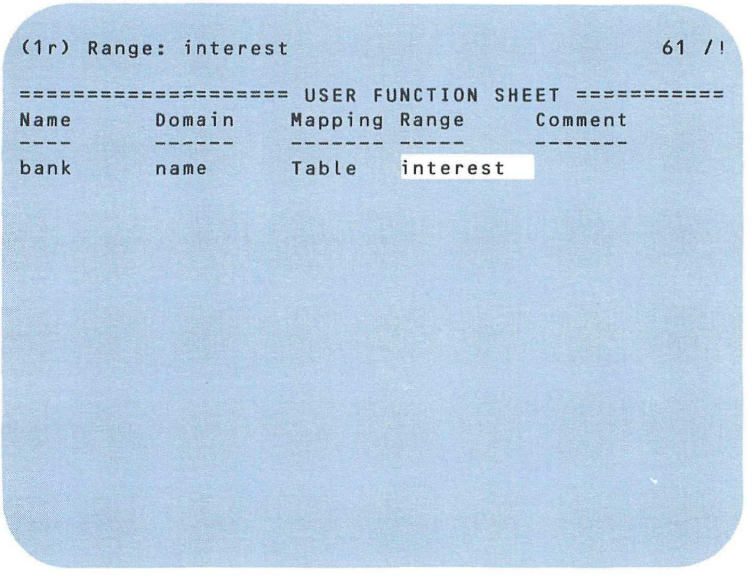
Numeric values may be automatically generated by assigning a numeric value to the first element and the last element, and using the Action command (!Y) to calculate the intermediate values. Values may also be filled in as a result of the action of the Solve or Block option of the List command if the list is designated as an output list.

See the Mathematics section for further information concerning use of lists in solutions.

Values in a list corresponding with a domain or range list on a User Function subsheet are updated simultaneously on both List subsheets.

The User Function Sheet

The User Function Sheet contains a table of all functions defined by the user. A function can be defined as the relationship between two sets of values, the domain and the range. Each row can contain the description for one user function. There is a User Function subsheet displaying more information about each user function; the values for the domain and range are displayed on a User Function subsheet and two separate List subsheets. Items entered in fields on the User Function Sheet are updated simultaneously on corresponding fields on User Function subsheets.



Access:
The User Function Sheet can be displayed in the current window by typing =**F** or in a new window by typing /**WF** from a one-window display.

A User Function subsheet for a user function can be reached by typing the Dive command (**)**) with the cursor on any field on the function row.

The List subsheet associated with the domain or range of a user function can be reached from the User Function Sheet by placing the cursor on any field in the row and typing the Dive command

(›) twice, once into the User Function subsheet, and a second time into the List subsheet for either the domain or range list. The domain List subsheet can be reached from the User Function Subsheet by placing the cursor on the Domain List field or on any value in the Domain column and typing the Dive command (›). The range List subsheet can be reached from either the Range List field or any value field in the Range column.

Initial Cursor Position:
The Name field on the first row

Fields:
Each user function is described by a row containing five fields, located in columns under the labels in the Label line at the top of the sheet. The User Function Sheet fields are:

Name (n)
[Name Entry]

Accepts a unique user function name. A user function may remain unnamed. A user function must be defined before it can be used in equations or expressions. The function name cannot duplicate a built-in function name; built-in function names, unlike user function names, are case insensitive. For example, "func1" and "FUNC1" are two different user functions, while "sin" and "SIN" both refer to the same built-in sine function.

Domain (d)
[Name Entry]

Accepts a list name to designate the domain of the function. The domain contains the values for the argument of the function. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet and in the Domain List field on the corresponding User Function subsheet.

Any units for a domain list must be entered on the List Sheet or List subsheet for the domain list. Functional relationships are determined only for Display unit values.

Mapping (m)
[Required Option]
Default: Table

T Table Option
S Step Option
L Linear Option

Defines the interpolation relationship between the domain and range values. The Table lookup option is automatically displayed in the field as soon as an item is entered in any field on the row. It may be changed to any of the other options but the field cannot be blanked.

Table lookup and linear interpolation are possible in either direction between the domain and range. Step functions can only be interpolated from the domain to the range.

Values supplied as function arguments for user functions must fall within the bounds of the specified list. Both lists should be the same length because the additional elements in the longer list are ignored.

If the exact value of the function argument for the user function is in the specified list, it maps directly to the corresponding value in the other no matter which mapping option is selected. If the exact value of the function argument is not a value in the specified list:

- table lookup returns an error,
- linear interpolation calculates a value in between the values in the corresponding list,
- the step function chooses the value in the range list corresponding to the previous value in the domain list.

Range (r)
[Name Entry]

Accepts a list name to designate the range of the function. The range list contains the function values. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet and in the Range List field on the corresponding User Function subsheet.

Any units for a range list must be defined on the List Sheet or List subsheet for the range list. Functional relationships are determined only for Display unit values.

Comment (c)
[Comment Entry]

Accepts a comment or description of the user function. Any characters entered into this field are ignored by the program in reaching solutions.

User Function Subsheets

Each User Function subsheet contains a detailed description of one function defined on the User Function Sheet. There is one User Function subsheet for each function on the User Function Sheet whether named or not. Items entered in fields on a User Function subsheet are updated simultaneously on corresponding fields on the User Function Sheet.

Each User Function subsheet is associated with two List subsheets, one for the domain and one for the range, containing the same values as the User Function subsheet. Values entered in the domain and range lists on a User Function subsheet are automatically entered in the Value fields of the corresponding List subsheets.

Domain and range values should be expressed in Display units. Functional relationships are determined strictly by the Display unit values.

```
(10r) Range: 15.5                                        61 /!

==================== USER FUNCTION: bank ===========
Comment:
Domain List:                    name
Mapping:                        Table
Range List:                     interest
Element  Domain        Range
-------  ------        -----
1        'first_town   12.5
2        'new_town_b   14
3        'old_town_b   18
4        'mid_town_b   13.5
5        'little_tow   16
6        'big_town_b   19
7        'tiny_town_   14.5
8        'huge_town_   18
9        'small_town   17.5
10       'town_bank    15.5
```

Access:
A particular User Function subsheet is accessed by placing the cursor anywhere on the row on the User Function Sheet for the function to be examined and typing the Dive command (⟩). Typing the Return command (⟨) returns to the User Function Sheet.

The List subsheet associated with the domain can be reached by placing the cursor on the Domain List field or on any value in the Domain column, and typing the Dive command (⟩). The range List subsheet can be reached from the Range List field or any value in the Range column.

Initial Cursor Position:
The first Domain field

Fields:
The user function defined on a User Function subsheet is described by four header fields at the top of the sheet, with a field label and a field on each line. Each function element is described by a row in the Field Area containing three fields in columns under the Label line. The domain and range values are identified by an element number automatically generated by the program on the left side of the sheet as values are entered in either column. Fields appearing on both a User Function subsheet and the User Function Sheet behave in the same way unless otherwise noted. The User Function subsheet fields are:

Comment: (c)
[Comment Entry]

Same as the Comment field on the User Function Sheet.

Domain List: (f)
[Name Entry]

Same as the Domain field on the User Function Sheet. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet and in the Domain field on the User Function Sheet.

Mapping: (m)
[Required Option]
Default: Table

T Table Option
S Step Option
L Linear Option

Same as the Mapping field on the User Function Sheet, except that diving to a User Function subsheet for an unnamed function assigns the default Table mapping option automatically.

Range List: (t)
[Name Entry]

Same as the Range field on the User Function Sheet. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet and in the Range field on the User Function Sheet.

Element
[Restricted Output]

Element numbers are automatically assigned as domain and range values are entered. The cursor cannot be moved to the Element column.

Domain (d)
[Value Entry/Output]

Assigns a numeric or symbolic value to a domain list element. A domain value entry can be any valid numeric value or expression, or a symbolic value.

Numeric values may be automatically assigned by defining the first and last values for the domain list and using the Action command (!Y) to calculate the intermediate values. The cursor must be placed on a field in the Domain column. Values may also be filled in as a result of the action of the solvers **/L!** and **/LB** if the list is designated as an output list.

The values for the domain list may be entered in the Domain column on the User Function subsheet or on the corresponding List subsheet. They are automatically entered on the other subsheet.

See the Overview section for further information concerning valid value entries.

Range (r)
[Value Entry/Output]

Assigns a numeric or symbolic value to a range list element. A range value entry can be any valid numeric value or expression, or a symbolic value.

Numeric values may be automatically assigned by defining a first and last value for the range list and using the Action command (!Y) to calculate the intermediate values. The cursor must be placed on a field in the Range column. Values may also be filled in as a result of the action of the solvers **/L!** and **/LB** if the list is designated as an output list.

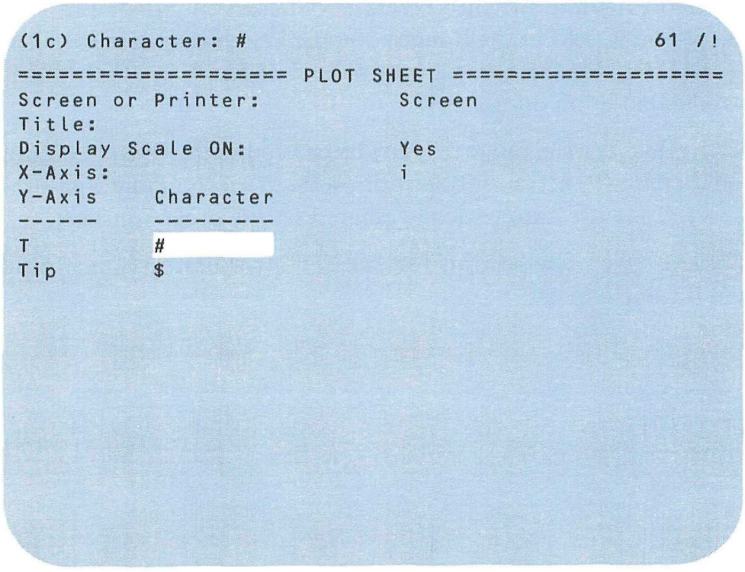
The values for the range list may be entered in the Range column on the User Function subsheet or on the corresponding List subsheet. They are automatically entered on the other subsheet.

See the Overview section for further information concerning valid value entries.

The Plot Sheet

The Plot Sheet contains information necessary to plot values in one list against the values in all other specified lists. Each row in the Field Area can contain the description for one list.

A plot is produced from the Plot Sheet with the Action command (!).



Acesss:
The Plot Sheet can be displayed in the current window by typing =**P** or in a new window by typing /**WP** from a one-window display.

A List subsheet for lists described on the Plot Sheet can be reached by placing the cursor on the X-Axis field or any field on a Y-axis row and typing the Dive command (›).

Initial Cursor Position:
The Screen or Printer field

Fields:
The plot defined on the Plot Sheet is described by four header fields at the top of the sheet, with a field label and a field on each line. Each Y-axis list on the plot is described by a row in the Field Area containing two fields in columns under the Label line. The Plot Sheet fields are:

Screen or Printer: (s)
[Required Option]
Default: Screen

S Screen Option
P Printer Option

Designates the destination for the plot output. If the field is set to "Screen", the plot is displayed on the screen in a format compatible with the video screen size. The printer definitions on the Global Sheet are ignored. To return to the previous display after the plot, the Enter key must be pressed.

If the field is set to "Printer", the plot is formatted using the printer options set on the Global Sheet and sent to the specified printer device or filename. The output is displayed on the screen while it is being sent to the printer or file. The Printed Page Width and Printed Page Length fields on the Global Sheet can be used to increase or decrease the size of the printed plot.

If the Printer Device or Filename field on the Global Sheet is blank, the output is only displayed to the screen, with the specified printer formats. The Enter key is required to return to the previous display after the plot.

The Break key cancels the plot at any point, produces an error signal and returns immediately to the previous display.

Title: (t)
[Comment Entry]

Specifies an optional plot title to be displayed at the bottom of a plot. If specified, a title decreases the plot length by two lines. If the title is wider than the screen or the Printed Page Width field on the Global Sheet, it is truncated.

Display Scale ON: (d)
[Required Option]
Default: Yes

Y Yes Option
N No Option

Controls whether an approximate scale is displayed on the axes of the plot. If the field is set to "Yes", a scale of numbers is printed along the X-axis and Y-axis to indicate an intermediate set of numbers evenly distributed between the determined end points. The numbers on the X-axis are determined from the largest and smallest values in the X-axis list. The numbers on the Y-axis are determined from the largest and smallest values in all Y-axis lists. If the difference between the largest and smallest values for either axis is greater than 1E120, the word " = ERROR = " displays instead of the scaling numbers on the axes.

The length of each axis and the scale numbers displayed are affected by the size of the plot. The size of the plot is determined by screen size when the "Screen" option is selected, or by the Printed Page Length and Printed Page Width settings on the Global Sheet when the "Printer" option is selected.

If the field is set to "No", the X-axis and Y-axis are displayed with tic marks and no scaling numbers, allowing a slightly larger image for the plot.

X-Axis: (x)
[Name Entry]

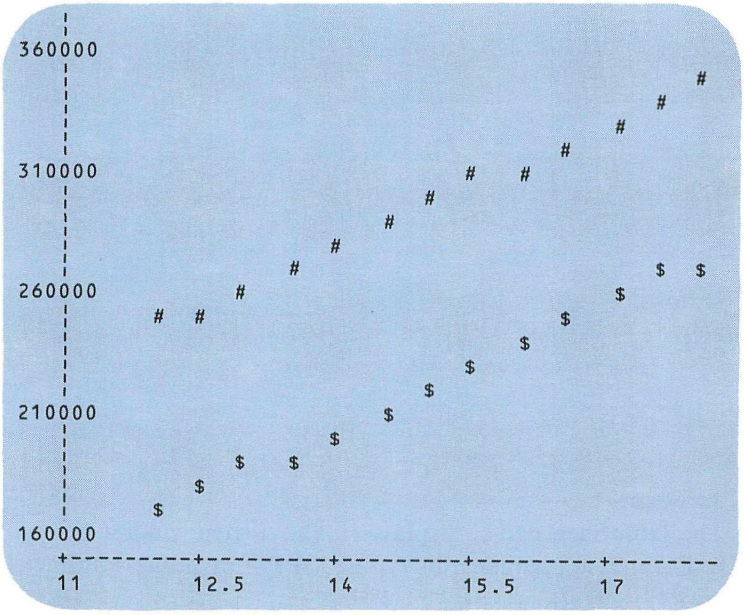
Designates a single list as the values for the horizontal coordinates (abscissa values). If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet. An X-axis list name must be specified to produce a plot.

Y-Axis (y)
[Name Entry]

Designates the lists to be used as vertical coordinates (ordinate values), which in conjunction with the abscissa values determine the points of the plot. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet. If any specified Y-axis list is longer than the X-axis list, only the corresponding number of values is plotted. At least one Y-axis list name must be specified in order to plot values.

Character (c)
[Option]
Default: *

Designates the character to be used to plot the points associated with the Y-axis list specified on the same row. The default character is automatically supplied by the program as Y-axis list names are entered. It can be changed to any other displaying character or blanked to display nothing. A new character is accepted without pressing the Enter key. Command characters (? / = : ;) ⟨ !) may only be used if edited into the field with the Edit command (⊡). For overlapping plot points, only the character for the last list plotted is displayed.



Plot generated from Plot Sheet shown above.

The Table Sheet

The Table Sheet contains information necessary to display a table of list values. A table can be defined as a display or printout of designated values from all specified lists. Each row in the Field Area can contain the description for one list.

A table is produced from the Table Sheet with the Action command (!).

```
(1w) Width: 10                                          61 /!

==================== TABLE SHEET ====================
Screen or Printer:              Screen
Title:                          Sample Table
Vertical or Horizontal:         Vertical
List       Width First Header
----       ----- ----- ------
i          10    1     Rate
A          18    1     Monthly Payment
T          18    1     Total Payments
```

Access:
The Table Sheet can be displayed in the current window by typing =**T** or in a new window by typing /**WT** from a one-window display.

The List subsheet for a list described in the Field Area can be reached by placing the cursor on any field in the row and typing the Dive command (›).

Initial Cursor Position:
The Screen or Printer field

Fields:
The table defined on the Table Sheet is described by three header fields at the top of the sheet, with a field label and a field on each line. Each list on the table is described by a row in the Field Area containing four fields in columns under the Label line. The Table Sheet fields are:

Screen or Printer: (s)
[Required Option]
Default: Screen

S Screen Option
P Printer Option

Designates the destination for table output. If the field is set to "Screen", the table is displayed on the screen in a format compatible with the video screen size. The printer definitions on the Global Sheet are ignored. To see a subsequent page of information or to return to the previous display after the last page of a table, the Enter key must be pressed.

If the field is set to "Printer", the table is formatted using the printer options set on the Global Sheet and sent to the specified printer device or filename. The output is displayed on the screen while it is being sent to the printer or file without waiting for the Enter key to go from page to page, or return to the previous display.

If the Printer Device or Filename field on the Global Sheet is blank, the output is only displayed to the screen, with the specified printer formats. The Enter key is required to see a subsequent page of information or to return to the previous display after the last page of a table.

The Break key cancels the table at any point, produces an error signal, and returns immediately to the previous display.

Title: (t)
[Comment Entry]

Specifies an optional title to be displayed at the top of a table. If specified, a title decreases the table page length by two lines. If the title is wider than the screen or the Printed Page Width field on the Global Sheet, it is truncated. The title is printed again at the top of each page or segment of lists.

Vertical or Horizontal: (v)
[Required Option]
Default: Vertical

V Vertical Option
H Horizontal Option

Designates the orientation of the table.

Vertical tables organize the values in columns under headers printed across the top of the page. If a vertical table has more lists than can be printed on a single page, all values for the lists on the first page are printed before printing any other lists.

Horizontal tables display the headers to the left of the values. If a horizontal table has more values in any list than can be printed on a single page, the same number of values for all lists are printed before printing additional values.

Lists of varying lengths may be printed on the same table. The length allowed for all lists is determined by the longest list.

List (l)
[Name Entry]

Assigns the lists of values to be displayed in rows or columns. If the list name does not previously exist, it is inserted at the bottom of the Name column on the List Sheet. If the list has no values, it is printed as a blank column under its header. A blank row inserted in the list of list names is ignored when a table is produced. At least one list must be specified with assigned values to produce a table.

Width (w)
[Required Integer Entry]
Default: 10

Designates the display width for the values in the list. There is a one-character space left automatically between columns.

A default width of 10 is automatically supplied by the program whenever an item is entered in any other field on the row. The width setting may be changed to any integer from 3 to the width of the screen or printer being used (up to 128), but the field cannot be blanked.

Each column on a vertical table prints according to the specified width settings. If more than one width value is specified for a horizontal table, the largest width value is used.

If a width value is supplied on a blank row, the default first element value is automatically entered by the program in the First field. A line containing only a width value and a first element value prints a blank column of the specified width within the table.

The number of decimal places displayed can be roughly approximated by controlling the width of the display field. In small column widths, wide numbers are displayed in scientific notation or rounded off.

First (f)
[Required Integer Entry]
Default: 1

Assigns the element of the list to display as the first element on the table. All list values are displayed from the specified element to the end of the list. The element number 1 is automatically supplied by the program whenever an item is entered in any other field on the row. It may be changed to any integer from 1 to the total number of elements in the list, but the field may not be blanked. If a first element value is supplied on a blank row, the default width value is automatically entered by the program.

Header (h)
[Comment Entry]

Specifies an optional label to display with values for a list on the same row. On vertical tables, headers are displayed above columns of values and truncated, if necessary, to fit within the specified width for each column. Care should be taken to begin headers for vertical tables with distinguishing information. If the Page Breaks ON field on the Global Sheet is set to "Yes", headers are repeated at the top of each page. If the Page Breaks ON field is set to "No", headers appear only at the top of new lists.

On horizontal tables, headers are displayed to the left of values. The first column of values is displayed to the right of the longest label in the table plus a single space. Care should be taken to make headers for horizontal tables shorter than the width of the

screen or the printed page width being used, minus at least the width of one column. On horizontal tables, headers are repeated for each segment of lists whether page breaks are on or off.

If no header is specified, the list name is used as the header. A header can be assigned to a row or column with no list name to print a label by itself.

Sample Table

Rate	Monthly Payment	Total Payments
12	800.4503481	240135.1044
12.5	828.6691435	248600.7431
13	857.154831	257146.4493
13.5	885.8901109	265767.0333
14	914.8583926	274457.5178
14.5	944.0438159	283213.1448
15	973.4312644	292029.3793
15.5	1003.006378	300901.9135
16	1032.755554	309826.6661
16.5	1062.665934	318799.7801
17	1092.725403	327817.621
17.5	1122.922573	336876.7719
18	1153.246757	345974.0271
18.5	1183.687954	355106.3862
19	1214.236823	364271.047

Table generated from Table Sheet shown above.

TK!Solver Command Chart

Solvers:

Direct Solver

To use the Direct Solver type / ! on any sheet or ! on the Rule or Variable Sheet.

Iterative Solver

To use the Iterative Solver, determine which variables require Iterative solution. For each such variable:

1. Type G in Status field
2. Enter a guess value

Then solve as for Direct Solver.

Solve option of List Command

The Solve option of the List command uses lists of input values to create lists of output, solving with either solver.

To use the Solve option of the List Command:

1. Type L in Status field of each variable associated with a list.
2. Give an arbitrary value to each variable associated with an input list. All other lists assumed output.
3. If list name not same as variable name, enter list name on Variable subsheet.
4. Solve by typing / L!.

Cursor movement:

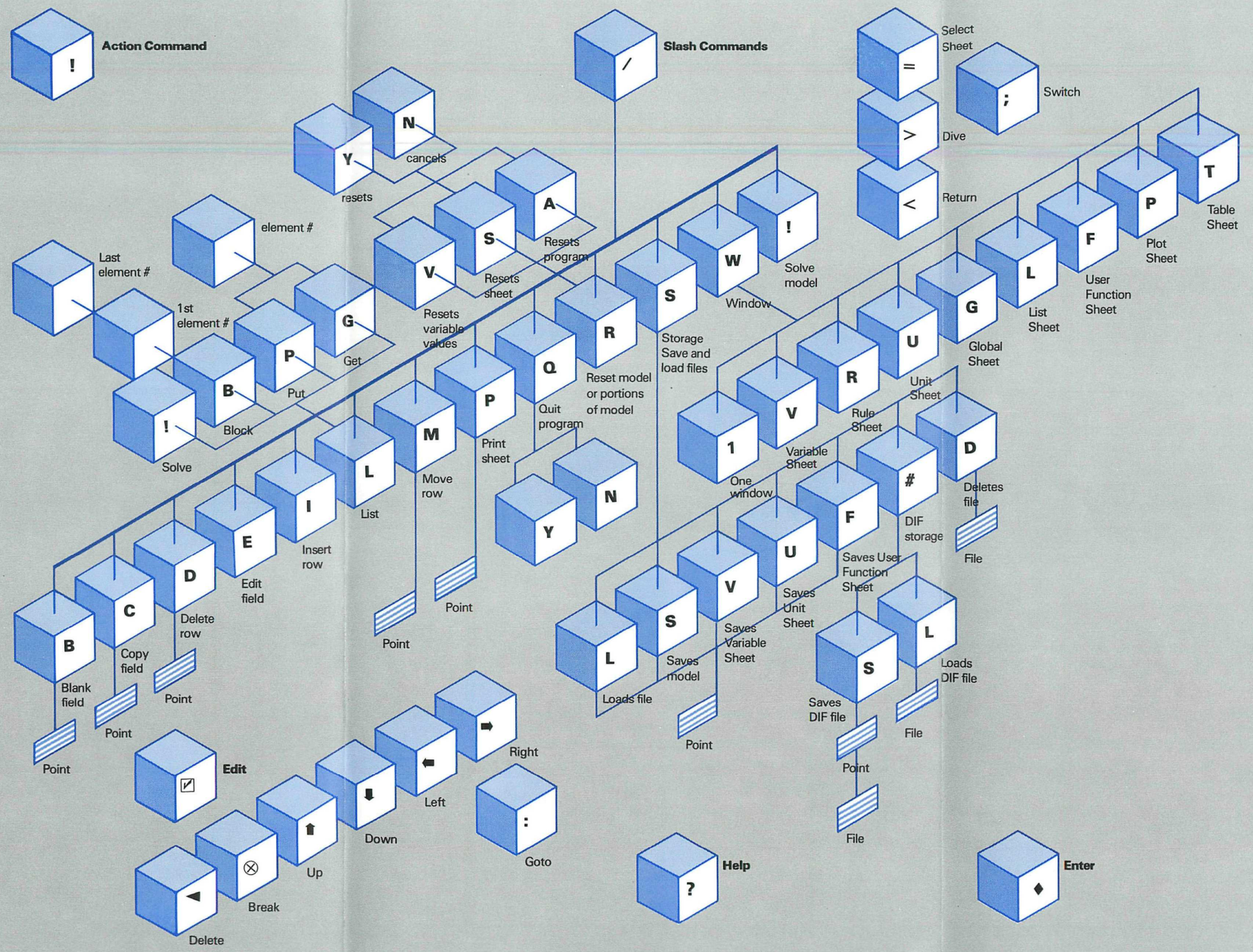
- ➡ Moves one field to right
- ⬅ Moves one field to left
- ⬆ Moves one row up
- ⬇ Moves one row down
- : Goto command moves cursor on sheet

Edit features

- ☑ edits unentered text
- /E edits entered text

Enter characters

- ◄ Deletes character to left
- ➡ Moves one character to right
- ⬅ Moves one character to left
- ⬆ Moves to beginning of text
- ⬇ Moves to end of text
- ⊗ Restores text state before edit



! Action Command

Variable Sheet: Solves model
Rule Sheet: Solves model
Plot Sheet: Produces plot
Table Sheet: Produces table
List Subsheet: Fills List
User Function Subsheet: Fills range and domain lists
Unit Sheet: No action
Global Sheet: No action
List Sheet: No action
User Function Sheet: No action

Sheet commands:

= **Select Sheet Command**
- **V** Variable Sheet
- **R** Rule Sheet
- **U** Unit Sheet
- **G** Global Sheet
- **L** List Sheet
- **F** User Function Sheet
- **P** Plot Sheet
- **T** Table Sheet

> **Dive Command**
Displays subsheet of current sheet if one exists.

< **Return Command**
Displays sheet subsheet was entered from

; **Switch Command**
Moves cursor between two displayed windows

? Help

Options:

?	Displays Help instructions
topic ♦	Displays info on *topic*
topics ♦	Displays list of Help topics
♦	Exits Help

Point Prompt to point cursor
Move cursor by:
Arrow keys
Goto command
Switch command

File <filename> ♦ selects file
➡ permits directory scrolling
♦ selects file from directory scrolling
For save, if file exists, prompts for confirmation before overwriting

Copyright © 1983 Software Arts, Inc.

P/N 200-046

Commands

There are two kinds of commands in the TK!Solver program: special computer keys and commands that use one or more character keys.

Cross-references to both command names and command usage are provided in the Contents section at the beginning of the Reference Manual.

Special keys are described by the names and symbols employed in this manual. For more specific information about the keys to press on your keyboard, see the Hardware Reference section.

Special keys are described for their actions in various environments:

Arrow keys	⬅ ➡ ⬆ ⬇
Enter key	◆
Break key	⊗
Backspace key	◀

Commands are alphabetized by their command names:

Action	!
Blank (field)	/B
Copy (field)	/C
Delete (row)	/D
Dive (subsheet)	>
Edit	◪
Edit Field	/E
Goto (field)	:
Help	?
Insert (row)	/I
List	/L
Move (row)	/M
Print (sheet)	/P
Quit	/Q
Reset	/R
Return (subsheet)	<
Select (sheet)	=
Solve	/!
Storage	/S
Switch (window)	;
Window	/W

Commands

Some commands use a single keystroke and act immediately. These commands include the Arrow keys, the Dive (>), Return (<), and Switch (;) commands, and the Solve, Plot, and Table actions of the Action command (!). Of these commands, the Break key can only cancel the options for the Action command.

Some commands, called extended commands, use more than one keystroke. The extended commands in the TK!Solver program are the Select command (=), the Goto command (:), the Fill action of the Action command (!), and any command beginning with a slash (/). The Break key can be used at almost any point to cancel an extended command.

Other commands call program facilities and have their own system of user actions and responses. The Edit command (◪) and the Edit Field command (/E) call the Editor. The Help command (?) calls the Help Facility. The Break key can be used to exit from these environments.

Certain extended commands act immediately and require no further user response. These commands include the Insert (/I) and Solve (/!) commands.

Certain extended commands prompt the user to point to a final cursor position and press the Enter key (♦). These commands include the Blank (/B), Copy (/C), Delete (/D), Move (/M), Print (/P), and DIF Save option (/S#S) of the Storage command.

Certain extended commands prompt for a selection of options on the Prompt/Error line. These commands include the List (/L), Reset (/R), Select (=), Storage (/S), and Window (/W) commands.

Certain commands require confirmation with Y (Yes) or N (No). These commands include the Fill action of the Action (!) command, some places in the Help Facility (?), the Quit (/Q), and Reset (/R) commands, and deleting or overwriting with the Storage command (/S).

Certain commands require user responses to be typed on the Prompt/Error line and completed by pressing the Enter key. These commands include the Goto (:), List (/L), and Storage (/S) commands, and prompts for topics in the Help Facility (?). The Edit command (◪) can be used during these commands prior to

the final Enter key to edit user responses on the Prompt/Error line.

The Storage command (/S) employs Directory Scrolling on the Prompt/Error line to select a filename that exists.

Arrow Keys

Command Mode:

The Arrow keys move the cursor from its current position one field in the direction of the arrow:

⬅	one field left
➡	one field right
⬆	one row up
⬇	one row down

Entry Mode:

The Arrow keys enter any data typed into the current field and move the cursor to the next field in the direction of the arrow. If the item is invalid for the current field, the user is automatically placed in the Editor. An error message displays on the Prompt/Error line and the cursor remains on the same field.

The Editor:

The Left Arrow key (⬅) moves the cue one character to the left.

The Right Arrow key (➡) moves the cue one character to the right.

The Up Arrow key (⬆) moves the cue to the beginning of the text.

The Down Arrow key (⬇) moves the cue to the end of the text.

Directory Scrolling:

The Right Arrow key (➡) is used to display a filename with the Directory Scrolling feature of the Storage command (/S). It can be used alone, or after a partial file specification to select all files on a diskette that correspond to the particular Storage command option. (See the Storage command and the Hardware Reference section for more information.) Once one filename is displayed, the Right Arrow key can be used to display each subsequent filename until the one to be selected appears or until there are no more matching filenames. In systems with a hierarchical directory structure, the names of subdirectories are displayed after the last filename.

♦ Enter Key

Program:
The Enter key (♦) starts the program (or stops it in the event of insufficient memory or a disk error) from the initial screen.

The Enter key also restarts the program after a disk or printer error once the program has begun.

Entry Mode:
The Enter key enters the characters typed into the current field as permanent data, if they consititute a valid item for the current field. If the item is invalid for the current field, the user is automatically placed in the Editor and an error message is displayed on the Prompt/Error line.

Command Mode:
The Enter key completes the pointing commands including the Blank (/B), Copy (/C), Delete (/D), Move (/M), and Print (/P) commands, and the DIF Save option of the Storage command (/S#S).

The Enter key completes any user response typed or edited on the Prompt/Error line for the Goto (:), List (/L), and Storage (/S) commands and performs the appropriate action.

The Editor:
The Enter key enters the characters typed or edited in the current field as permanent data, if they constitute a valid item for the current field. If the item is invalid for the current field, the user remains in the Editor and an error message is displayed on the Prompt/Error line.

The Help Facility:
The Enter key enters any topic name typed or edited on the Prompt/Error line and searches for the appropriate topic. It can also be used alone instead of entering a topic name to leave the Help Facility.

Plots, Tables, or Prints:
The Enter key displays the following page of a multiple-page table or print displayed only to the screen. The Enter key also returns the user to the previous display after a plot or the last screen of a table or print displayed only to the screen.

Directory Scrolling:
The Enter key selects the current filename for use with the options of the Storage command (/S).

⊗ Break Key

Program:
The Break key (⊗) cancels the program at the initial screen.

The Break key cancels the output action and returns to the program after a disk or printer error once the program has begun.

Pressing the Break key twice cancels the program after a program disk error once the program has begun.

Entry Mode:
The Break key restores the previous item any time before the Enter key is pressed. After the Enter key, the Break key has no effect.

The Editor:
The Break key restores the most recent item any time before the Enter key enters a new item.

Command Mode:
The Break key cancels a command any time before the last keystroke of an extended command.

The Break key interrupts the Help Facility and immediately returns the user to the previous display.

The Break key returns the user to the previous display before the last page of a multiple-page table or print displayed only to the screen.

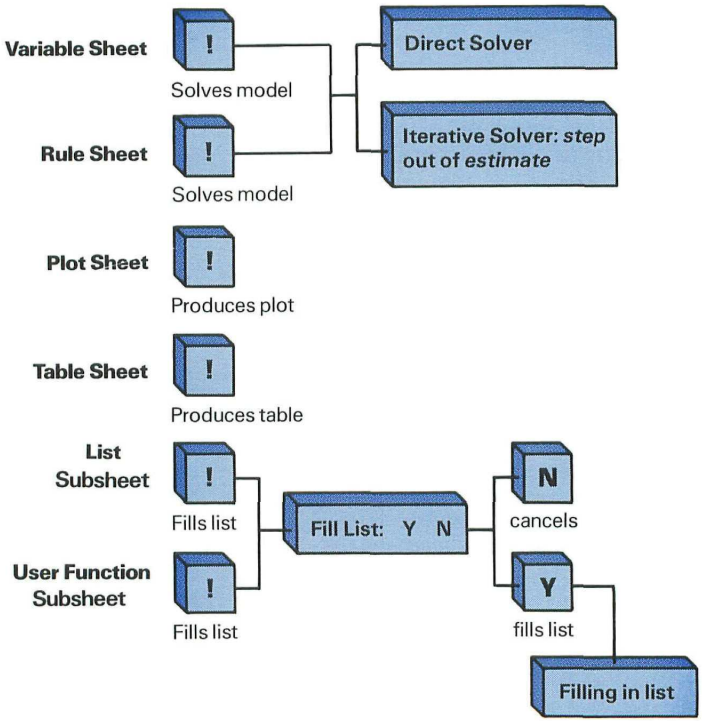
◄ Backspace Key

Entry Mode, Command Mode, the Editor or the Help Facility:
The Backspace key (◄) deletes the character to the left of the cue any time before the Enter key is pressed.

! Action Command

The Action command performs the action appropriate to the current sheet, including model solution, list value fill-in, plot or table display.

! Perform action



Description:

Solve Action (from the Variable or Rule Sheets)

Typing the Solve action (!) solves the currently defined model by calling the Direct Solver or the Iterative Solver, as required by the model. The cursor may be positioned on any field on the Variable Sheet or Rule Sheet.

The Iterative Solver is called automatically if either:

- a G option is set for one or more variables in the Status field on the Variable Sheet, or

- the Direct Solver fails, the Automatic Iteration ON field on the Global Sheet is set to "Yes", and values are supplied for one or more variables in the First Guess fields of their Variable subsheets.

During solution, a status message including the name of the Solver being used is displayed on the Prompt/Error line:

`Direct Solver`

or

`Iterative Solver:` *step* `out of` *estimate*

If the Iterative Solver is called, a count of the current iteration step and the estimated number of steps needed appears at the end of the status message as successive approximations are made. If the Intermediate Redisplay ON field on the Global Sheet is set to "Yes" (the default), the values at each iteration step are displayed in the Input and Output fields on the Variable Sheet during solution.

When solution is complete, results are displayed in the appropriate output fields of the Variable Sheet and dependent subsheets. If solution fails, error codes appear in the Status fields on the Rule and Variable Sheets. (See the appendix on Error Messages for more information.)

The Solution Indicator (!) in the upper righthand corner of the Status line disappears when the solvers stop and no errors have been found.

Fill Action (from List and User Function subsheets):
Typing the Fill action (!) calculates intermediate values in the current list. The cursor may be on any field on a List subsheet. On a User Function subsheet, the cursor may be on any field in the domain or range list to be filled in.

The block of elements to be filled in is determined by the last element in the list containing a specified value. The block must begin at Element 1, and both values must be numeric. The difference between the first value in the list and the last is evenly distributed over the intervening elements; values are filled in for all intervening elements, whether or not they contained previous values. If unit names have been assigned for the list, the filled in values appear in the Display unit.

The user is prompted for confirmation:

```
Fill List: Y N
```

Typing **Y** calculates and enters the values; typing **N** cancels the Fill action. The status message "Filling in list" is displayed while the values are being inserted.

Plot Action (from the Plot Sheet):
Typing the Plot action (!) displays a plot of the relationship between the values in one list to the values in all others specified on the Plot Sheet. The cursor may be positioned on any field on the Plot Sheet.

The values from the List subsheet specified in the X-Axis field are plotted against the values of all those specified in the Y-Axis column. Plot values are displayed in the Display units for the specified lists.

A one-page plot is displayed to the screen. It is also printed to the printer if the Screen or Printer field is set to **P** and the printer filename is specified for the appropriate printing device in the Printer Device or Filename field on the Global Sheet. When the screen option is set to **S** (the default) or the Printer Device or Filename field is blank, it is necessary to press the Enter key to return to the previous display from the plot.

Table Action (from the Table Sheet):
Typing the Table action (!) displays tables of values from the List subsheets for all lists specified on the Table Sheet. The cursor may be positioned anywhere on the Table Sheet. Table values are displayed in the Display units for the specified lists.

A one or more page table is displayed to the screen. It is also printed to the printer if the Screen or Printer field is set to **P** and the printer filename is specified for the appropriate printing device in the Printer Device or Filename field on the Global Sheet. When the screen option is set to **S** (the default) or the Printer Device or Filename field is blank, it is necessary to press the Enter key to display each page after the first, and then, after the last page, return to the previous display.

Usage Notes:

- For more information on using the Solve action, see the descriptions of the Solvers in the Mathematics section.
- For more information on using the Plot and Table actions, see the descriptions of the Plot and Table sheets in the Sheet section.

Errors:

- During the Fill action (! from List or User Function subsheets), typing ⊗ cancels the command. The computer produces an error signal, the Fill prompt disappears from the Prompt/Error line, and the command is cancelled. Any values already filled in remain on the screen.
- See the Overview and Mathematics sections and the appendix on Error Messages for more information about solution failures.

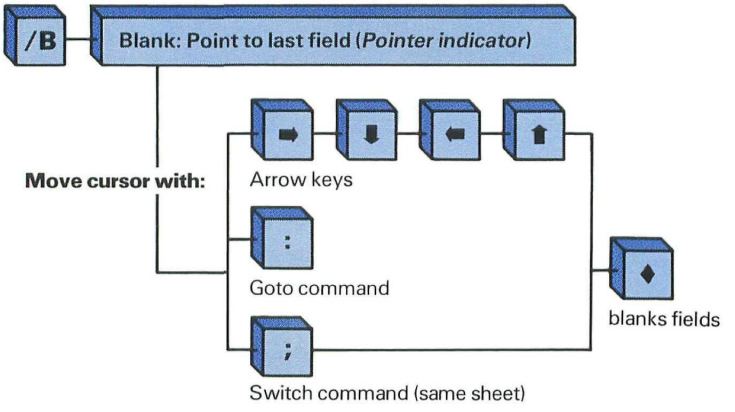
Related Commands:

- The Solve command (/!) solves the currently defined model from any sheet.
- The Solve (/L!) and Block (/LB) options of the List command apply the solvers to variables associated with lists.

/B Blank Command

The Blank command erases the current field item, or a column of items beginning with the current item.

/B Blank



Description:
Typing **/B** ♦ blanks the current field item. The Blank command is completed by pressing the Enter key. The cursor remains on the same field.

Typing **/B***pointing* ♦ blanks field items in the current column from the starting cursor position to the final cursor position in the same column. The program prompts for the last field to blank and indicates the starting cursor position on the Prompt/Error line:

```
Blank: Point to last field (Pointer Indicator)
```

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position as it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting cursor position. The Arrow keys may be used in any combination up or down with the Goto and Switch commands to reach the final cursor position on the same sheet. When the Blank command is completed by pressing the Enter key, the cursor returns to its starting position and the Blank command prompt disappears.

Usage Notes:

- The Goto command (:) Destination or Search options may be used to move the cursor towards the final cursor position. The Pointer Indicator is not displayed during the Goto subset of the Blank command.
- The Switch command (;) may be used to switch the cursor from one view of a sheet in one window to another view of the same sheet in the other window. Both windows must contain the same sheet prior to beginning the Blank command. This feature is useful in blanking long columns of items.
- The Blank command cannot be used on required option, required entry, or output fields; or on a subsheet displayed to the screen for a deleted variable, list, or user function.
- Items in the Field Area and Header Area of a sheet cannot be blanked with the same Blank command.
- The Blank command can be used to erase blankable field items in a block of rows containing both blankable and unblankable fields. A warning message is displayed.
- If the Display unit in a Unit field on the Variable or List Sheet is blanked, both the Display unit and the Calculation or Storage unit are cleared.
- If a value on a List subsheet, or both values in the same row for the Domain or Range fields on a User Function subsheet, are blanked, the corresponding element number is not displayed but the space is retained.
- The default value "*" in the Character field on the Plot Sheet is the only option field that may be blanked by editing in a space, so that the display of the corresponding list may be suppressed on a plot.
- If an item is blanked from an input field on the Variable Sheet or the Input Value field on a Variable subsheet, it is retained in memory and can be returned by typing an "I", "O", or "G" option in the Status field.

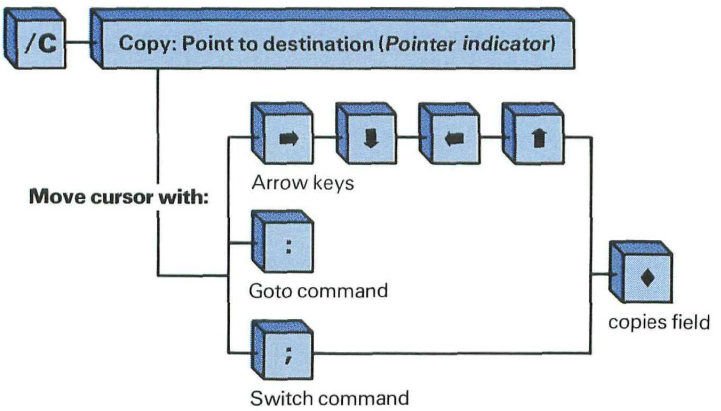
Related Command:

- The Delete command (/D) deletes entire rows.

/C Copy Command

The Copy command copies the current field item to the specified destination.

/C Copy field



Description:

Typing */Cpointing* ♦ copies the current field item to the indicated final cursor position on any field displayed on the screen. The Copy command copies the complete contents of the field, not only the portion of the item displayed in the field. The Copy command is completed by pressing the Enter key.

The program prompts for the destination for the copy and indicates the starting cursor position on the Prompt/Error line:

`Copy: Point to destination (Pointer Indicator)`

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position as it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting position. The Arrow keys may be used in any combination in any direction with the Goto and Switch commands to reach the final cursor position. When the Copy command is completed by pressing the final

Enter key, the cursor remains at the destination and the Copy command prompt disappears.

A field item can be copied from an entry, option, or output field to any entry or option field on the screen. The copied item replaces any previous item in the destination field. If a blank field is copied to another location, the destination field is erased of any previous contents. If the destination is an entry field and the copied item is invalid in the new position, an error message appears on the Prompt/Error line and the user is placed in the Editor. If the destination is an option field and the first character of the copied item is an invalid option, the valid options are displayed on the Prompt/Error line.

Usage Notes:

- The Goto command (:) Destination or Search options may be used to move the cursor to the final destination. The Pointer Indicator is not displayed during the Goto subset of the Copy command.
- The Switch command (;) may be used to switch windows so that an item can be copied from one sheet or subsheet to another, or from one view of a sheet or subsheet to another. Both sheet views must be displayed on the screen at the beginning of the command.
- The Copy command can be used to copy the contents of a short display field into a longer one to see more information.
- The Copy command cannot copy an item to an output field.

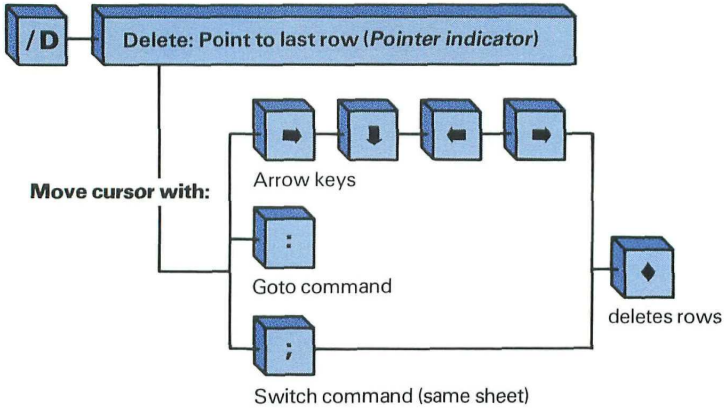
Related Command:

- The Move command (/M) moves the current row to the specified destination.

/D Delete Command

The Delete command deletes the current row, or a block of rows beginning with the current row.

/D Delete rows



Description:
Typing **/D** ♦ deletes the entire current row. The cursor may be positioned on any field of the row to be deleted. The Delete command is completed by pressing the Enter key. The cursor remains in the same position, and the rows below the deleted row are moved up.

Typing **/D***pointing* ♦ deletes rows from the current cursor position to the final cursor position on the same sheet. The cursor may be positioned on any field in either row. The program prompts for the last field to be deleted and indicates the starting cursor position on the Prompt/Error line:

`Delete: Point to last row (`*Pointer Indicator*`)`

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position at it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting cursor position.

The Arrow keys may be used in any combination in any direction with the Goto and Switch commands to reach the final cursor position on the same sheet. When the Delete command is completed by pressing the Enter key, the cursor returns to the starting cursor position, the rows below the deleted rows are moved up, and the Delete command prompt disappears. The cursor marks the item below the last deleted item in the same column.

Usage Notes:

- The Goto command (:) Destination or Search options may be used to move the cursor towards the final cursor position. The Pointer Indicator is not displayed during the Goto subset of the Delete command.
- The Switch command (;) may be used to switch the cursor from one view of a sheet in one window to another view of the same sheet in the other window. Both windows must contain the same sheet prior to the beginning of the Delete command. This feature is useful in deleting many lines.
- The Delete command can only delete rows in the Field Area of a sheet.
- If a row in the Field Area on a List subsheet or a User Function subsheet is deleted, any subsequent rows in the list are renumbered and moved up.

Related Command:

- The Blank command (/B) blanks field items.

> Dive Command

The Dive command displays a dependent subsheet for the current sheet or subsheet in the current window.

> Dive



displays subsheet

Description:
Typing > from certain sheets or subsheets replaces the display in the current window with the appropriate dependent subsheet. The dependent subsheet for a sheet or subsheet contains more detailed information about the variable, list, or function on the current line. When the subsheet appears, the cursor always moves to the initial cursor position for the subsheet.

From the Variable Sheet:
Typing > with the cursor on any field in a row on the Variable Sheet displays the Variable subsheet for the variable described in that row.

From a Variable subsheet:
Typing > with the cursor on any field on a Variable Subsheet with an item specified in the Associated List field displays the List subsheet for the associated list.

From the List Sheet:
Typing > with the cursor on any field in a row on the List Sheet displays the List subsheet for the list described in that row.

From the User Function Sheet:
Typing > with the cursor on any field in a row on the User Function Sheet displays the User Function subsheet for the user function described in that row.

From a User Function subsheet:
Typing > from a User Function subsheet displays the List subsheet for the domain or range list. To display the associated domain list, the cursor must be positioned on the Domain List field or on any value in the Domain column. To display the associated range list, the cursor must be positioned on the Range List field or on any value in the Range column.

From the Plot Sheet:
Typing > with the cursor on the X-Axis field on the Plot Sheet displays the List subsheet for the X-Axis list.

Typing > with the cursor on either Y-Axis field in a row in the Field Area of the Plot Sheet displays the List subsheet for the list described in that row.

From the Table Sheet:
Typing > with the cursor on any field in a row in the Field Area of the Table Sheet displays the List subsheet for the list described in that row.

Usage Notes:

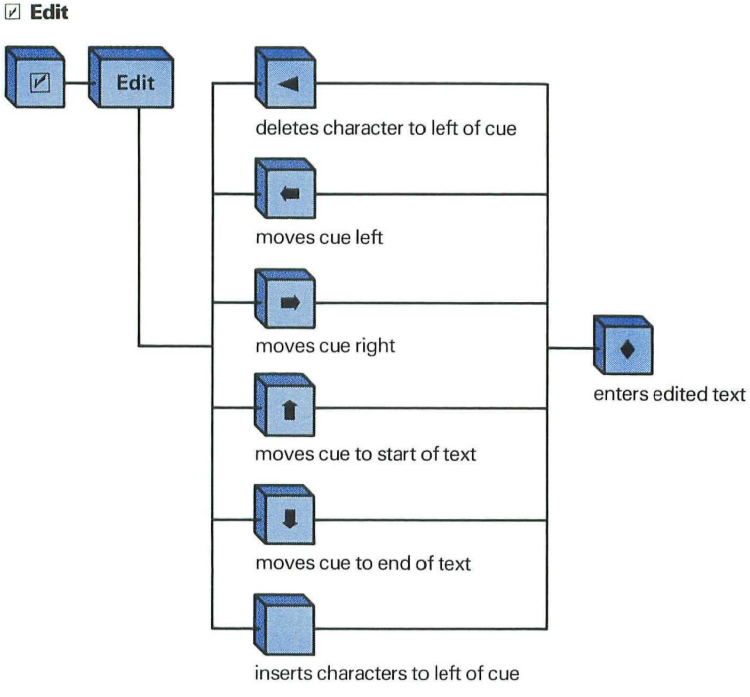
- The Dive command can be used twice in succession to display a List subsheet from a Variable Sheet or a User Function Sheet through the intermediate subsheet.
- There are no subsheets associated with the Rule, Unit, or Global sheets.
- The Return command (<) can be used to return to the previous sheet or subsheet, following the same path in reverse.
- It is possible to use the Select (=) or Window (/W) commands without using the Return command (<) first.

Related Commands:

- The Return command (<) returns the display of the previous sheet or subsheet to the current window after the Dive command (>).
- The Select command (=) displays the specified sheet in the current window.

Edit Command

The Edit command edits a user response on the Prompt/Error line or an unentered item in the current entry field.



Description:
Typing ◪ calls the Editor in the current field or on the Prompt/Error line. It can be called from any of the commands that require user response on the Prompt/Error line before the Enter key is pressed to complete the entry. It can also be called from Entry mode on the current entry field before the Enter key is pressed to enter the item. It can be used, as well, to edit an option, an entered item, or an empty field. When ◪ is typed, the Edit Indicator ["(Edit)..."] appears on the Prompt/Error line and the cue appears to the left of the first character in the item or the user response on the Prompt/Error line. In the Editor, the Arrow keys move the cue within the entry text:

⬅ Moves the cue left one character
➡ Moves the cue right one character
⬆ Moves the cue to the beginning of the text
⬇ Moves the cue to the end of the text

Characters may be typed as in Entry mode. Special keys perform particular actions in the Editor, similar to those in Entry mode:

◀ Deletes the character to the left of the cue
♦ Exits the Editor and enters the edited changes
⊗ Exits the Editor without changes

Pressing the Break key any time before the final Enter key when editing a user response cancels the command. Pressing the Break key when editing an item cancels the Editor session and restores the previous item, if any, to the field.

Usage Notes:
The Edit command ◪ can be used to edit the following user responses:

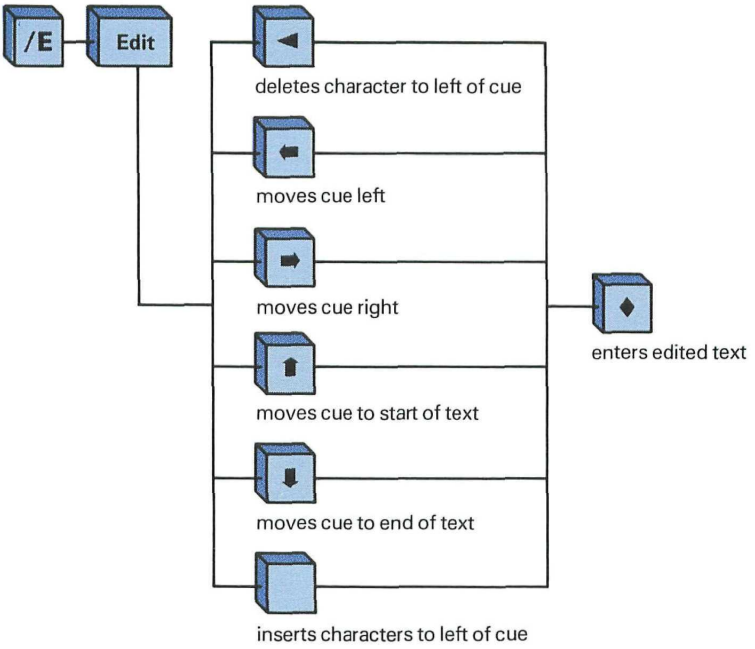
- Addresses or search items with the Goto command (:).
- Filenames for the Storage command (/S).
- Element numbers with the Block, Put, and Get options of the List command (/L).
- Topic names in the Help Facility.

Related Command:
- The Edit Field command (/E) edits the entered item in the current entry field.

/E Edit Field Command

The Edit Field command edits the entered item in the current entry field.

/E Edit field



Description:
Typing /E calls the Editor on the current entry field item. The Editor edits the complete contents of the field, not only the portion of the item displayed in the field. When /E is typed, the Edit Indicator ["(Edit)..."] appears on the Prompt/Error line and the cue appears to the left of the field item, at the beginning of the field. In the Editor, the Arrow keys move the cue within the text:

- ⬅ Moves the cue left one character
- ➡ Moves the cue right one character
- ⬆ Moves the cue to the beginning of the text
- ⬇ Moves the cue to the end of the text

Edit Field

Characters may be typed as in Entry mode. Special keys perform particular actions in the Editor, similar to those in Entry mode:

◀ Deletes the character to the left of the cue
♦ Exits the Editor and enters the edited changes
⊗ Exits the Editor without changes

If the Break key is pressed any time prior to pressing the Enter key, the previous item is restored to the field.

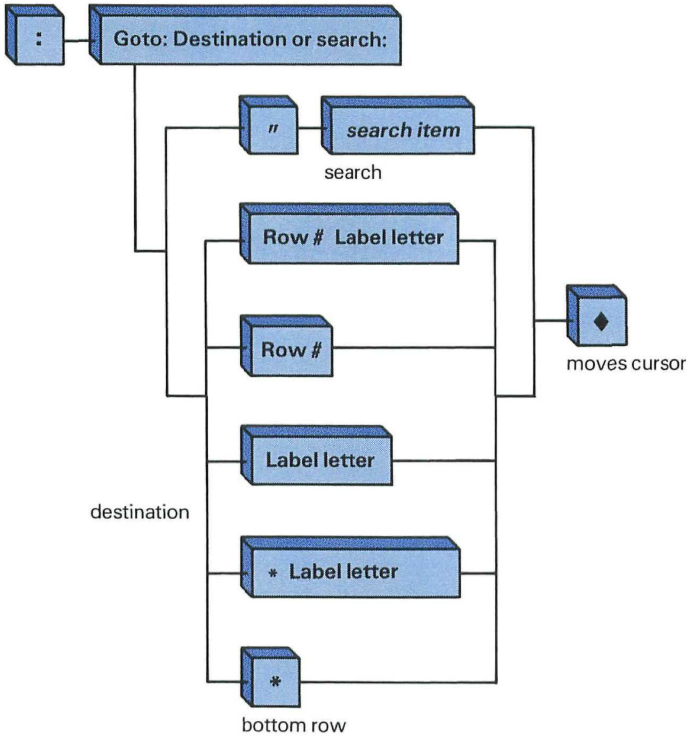
Related Command:

- The Edit command (◪) edits a user response on the Prompt/Error line or an unentered item in the current entry field.

: Goto Command

The Goto command moves the cursor to the specified field on the current sheet.

: Goto field



Description:

Destination Option (*address* ♦)

Typing the Destination option (:*address* ♦) moves the cursor from its current position to the specified destination on the same sheet.

An address is similar to the Position Indicator displayed on the Status line except that it need not be specified in full. The following ways of specifying an address are available:

row number label letter
row number
**label letter*
label letter
*

The complete address for a destination field is usually made up of a row number and label letter. A field in a Header Area is indicated simply by its label letter. (The label letter for every field is given in parentheses after its field label in the Sheet section.)

If only a row number is specified, the cursor moves up or down in the current column of the Field Area to the specified row. From the Header Area, the cursor moves to the first field on the specified row.

An asterisk can be substituted for the row number of the last row on a sheet containing an item (:* ♦). It may be used with or without a label letter.

If only a label letter is specified for a field in the Field Area, the cursor moves to the designated field on the same row.

The fields on the Global Sheet and a Variable subsheet can be located by typing their label letters or a sequential row number. On the Global Sheet, the blank lines between sections are included in the count.

The user response on the Prompt/Error line is completed by pressing the Enter key. It is possible to edit any characters typed on the Prompt/Error line before the Enter key by pressing ⌧.

Search Option ("*search item* ♦)
Typing the Search option (:"*search item* ♦) moves the cursor to the first field in the current column that displays the specified search item. Every character displayed in the field must be typed exactly as it appears to match, except that trailing blanks are ignored. If a match does not occur below the cursor, the column is searched from the top of the column back down to the current cursor position.

If the Search option is used without specifying a search item (:" ♦), the most recently specified item is reused.

The search item on the Prompt/Error line is completed by pressing the Enter key. It is possible to edit any characters typed on the Prompt/Error line before the Enter key by pressing ⊠.

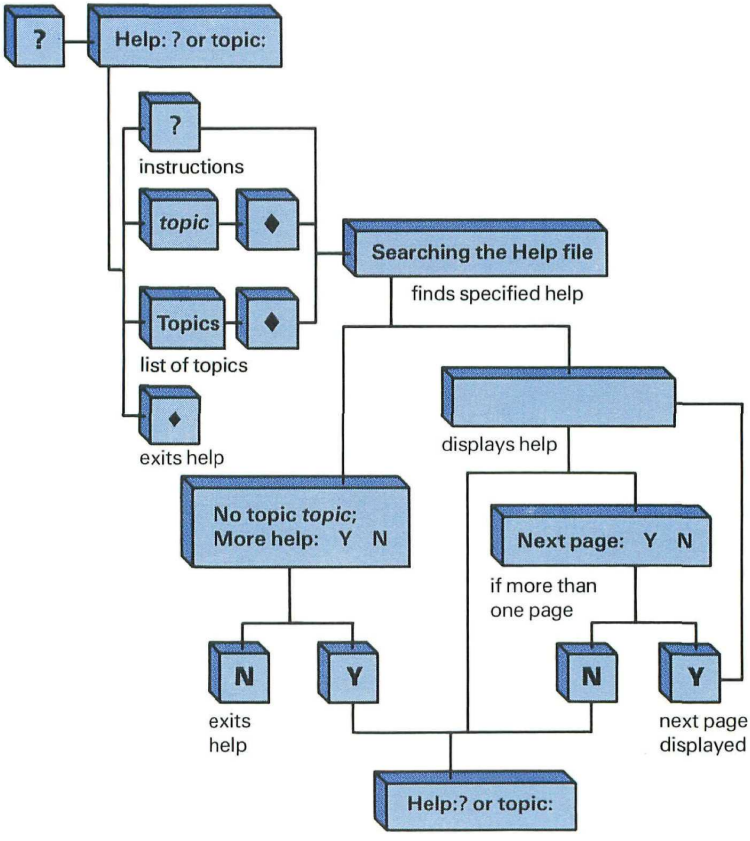
Usage Notes:

- The Goto command can be used to point in the Blank (/B), Copy (/C), Delete (/D), Move (/M), and Print (/P) commands, and the DIF Save option of the Storage command (/S#S).

? Help Command

The Help command calls the Help Facility.

? Help



Description:
Typing ? calls the Help Facility and displays the Help topic prompt on the Prompt/Error line of the current sheet display:

```
Help: ? or topic:
```

When a topic name is typed and entered with the Enter key in response to the Help topic prompt, a Status message "Searching the Help file" is displayed on the Prompt/Error line while the program searches for the first piece of related text. If a listing is

found for a topic, the Information Area is replaced with the Help Facility display: a line similar to a Sheet Title line displays the word "HELP" at the top of the window, and the first page for the requested topic displays in the window.

If the current topic displays in a single page, the Help topic prompt redisplays on the Prompt/Error line when the search is complete. If the current topic is more than a single page long, the following prompt appears on the Prompt/Error line:

```
Help: Next Page: Y N
```

Typing **Y** continues to the subsequent page; typing **N** returns directly to the Help topic prompt for another topic name.

If the specified topic does not exist, the following prompt appears on the Prompt/Error line:

```
No topic topic; More Help: Y N
```

Typing **Y** returns to the Help topic prompt for another topic name; typing **N** exits the Help Facility.

Instructions Option (?)
Typing the Instruction option (?) any time the Help topic prompt appears displays instructions on using the Help Facility, and any other topics related to the character "?". The Enter key is not used after the **?** as it must be after any other topic name.

Topic Option (*topic* ♦)
Typing a Topic option (*topic* ♦) any time the Help topic prompt appears displays all related text for the specified topic name. Typing **TOPICS** ♦ displays a list of Help Facility topics.

Exit Option (♦)
Typing ♦ any time the Help topic prompt appears exits the Help Facility and returns to the previous display and cursor position.

Pressing the Break key exits the Help Facility at any time.

Usage Notes:

- A topic name can be edited on the Prompt/Error line any time before the Enter key is pressed with the Edit command (⊡).
- Since Help Facility searches require time to perform and may take more than one page to display or may result in no topic, the next prompt should appear on the Prompt/Error line before another topic name is entered.
- If something is typed after the Help topic prompt and corrected to ? using the Backspace key, the Enter key must be pressed to enter the ? topic name.

/I Insert Command

The Insert command inserts a blank row at the current row.

/I Insert row



inserts row at current row

Description:
Typing /**I** inserts a new row at the current row. The cursor may be positioned on any field in the row below the one to be inserted. The cursor stays on the same field and all the rows below the new row move down.

Usage Notes:
- Insert command may only be used in the Field Area of a sheet.

/L List Command

The List command solves a model for lists of values associated with variables, or stores and retrieves list values.

Description:
The List command manipulates list values associated with variables. List command options are dependent on list information related to the current model and stored in the Variable, Rule, and List sheets, and their related subsheets. (For further discussion, see the Mathematics section.) The cursor may be positioned on any field on any sheet when using the List command.

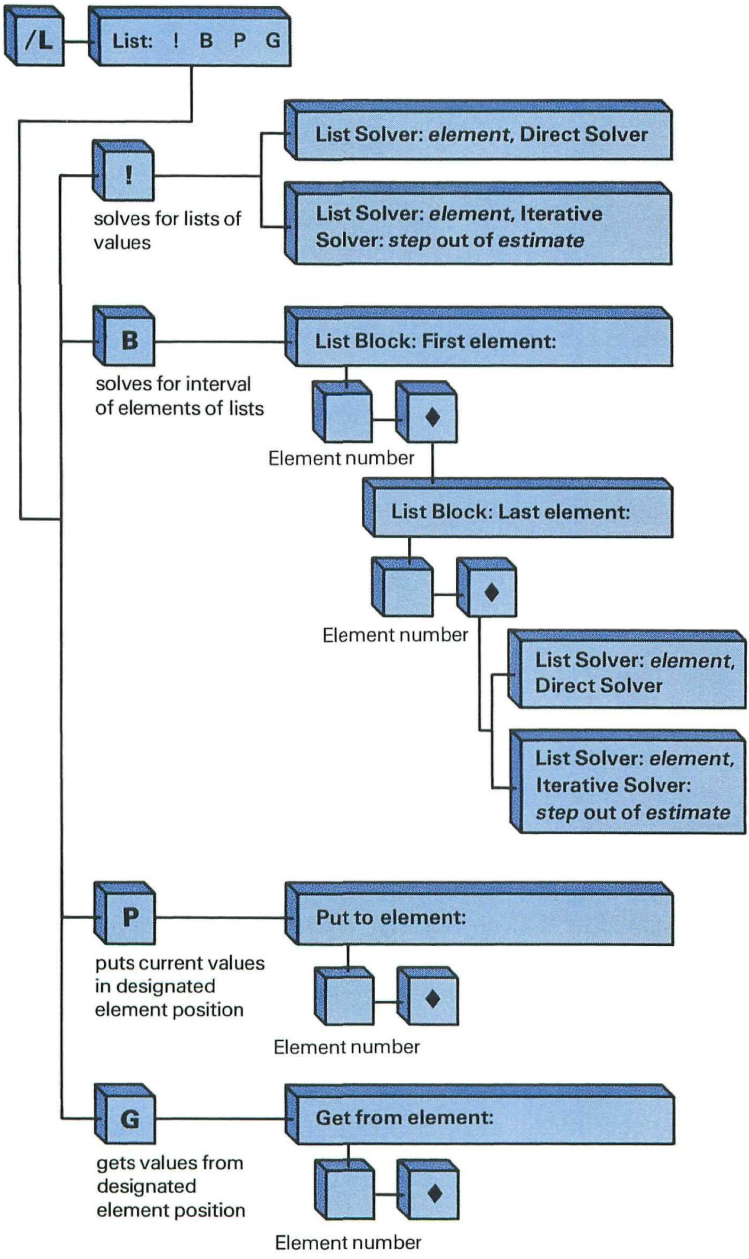
Solve Option (/L!)
Typing /**L!** applies either of the solvers to a model with at least one associated list. All variables to be associated with lists must display an "L" option in the Status field, and at least one initial input value must be assigned on the Variable Sheet before the List Solver can be called successfully.

The model is solved for all values in every list associated with a variable designated as an input or guess variable. Input lists are designated by assigning any value in the Input field for the associated variable. A list of guess values is designated by setting the "G" option in the Status field for any associated variable. The list of values associated with a guess variable are used as the first guess values for each element in the Iteration solution. The solution value for each element is assigned to the appropriate element number in the list as it is determined. If iteration is required and there are no values in a guess list, the First Guess value on the Variable subsheet for the associated variable is used for each element in the solution.

Output values are assigned during solution to any associated list that is not designated as an input list, and all previous values in those lists are replaced or blanked. The List Solver stops when the longest input list is exhausted; after solution, all output lists have the same number of elements as the longest input list, although some values may be blank. A blank value is produced in an output list for any value that cannot be evaluated in an input list.

If units are used, the Calculation unit of a variable should be the same as, or have a defined conversion to, the Storage unit of its associated list, or the List Solver produces unreliable results.

/L Manipulate lists



A Status message appears on the Prompt/Error line during solution indicating the current element number and the solver being used:

```
List Solver: element, Direct Solver
```

or

```
List Solver: element, Iterative Solver:
             step out of estimate
```

If the Intermediate Redisplay ON field on the Global Sheet is set to "Yes", the input and output values are displayed on the Variable Sheet as they are processed.

Block Option (/LB)

Typing **/LB***element* ♦ *element* ♦ applies either of the solvers to an interval of list values in a model with at least one associated list. The Block option is similar to the Solve option except that it solves for a subset of all lists.

The program prompts for the beginning and ending element numbers:

```
Block: First element:
Block: Last element:
```

The responses are typed on the Prompt/Error line and completed by pressing the Enter key.

The model is solved for each value within the specified block in every list associated with a variable designated as an input or guess variable, beginning with the first specified element number and ending with the last specified element number, inclusively, for all associated lists. If no first element is specified, the block begins with Element 1. If no last element is specified, the block ends with the last element in the longest input list.

Put Option (/LP)

Typing **/LP***element* ♦ assigns the current input and output values from the Variable Sheet for any list variables to the specified element number for all associated lists. The program prompts for the element number into which the values should be inserted:

```
Put to element:
```

The response is typed on the Prompt/Error line and completed by pressing the Enter key.

If the units for a variable and its associated list differ, the conversion is made from the Calculation unit for the variable into the Storage unit of the list before it is displayed on the List subsheet in Display units.

Get Option

Typing /**LG***element* ♦ assigns the values from the specified element number from the associated lists to any list variables on the Variable Sheet. It gets the values from the specified element number for all associated lists and replaces any current input and output values on the Variable Sheet. The program prompts for the element number from which to retrieve list values:

```
Get from element:
```

The response is typed on the Prompt/Error line and completed by pressing the Enter key.

If a variable associated with a list has not been designated an input or output list, the value is assigned to the Output column. If the units for a variable and its associated list differ, the conversion is made from the Storage unit for the list into the Calculation unit of the variable before it is displayed on the Variable Sheet in Display units.

Usage Notes:

- The Edit command (⊡) can be used to edit the user response to the Block, Put, and Get options any time before the Enter key is pressed.
- An expression that evaluates to an integer can be entered instead of an integer in response to the prompts for element numbers by the Block, Put, and Get options if it can be evaluated immediately.

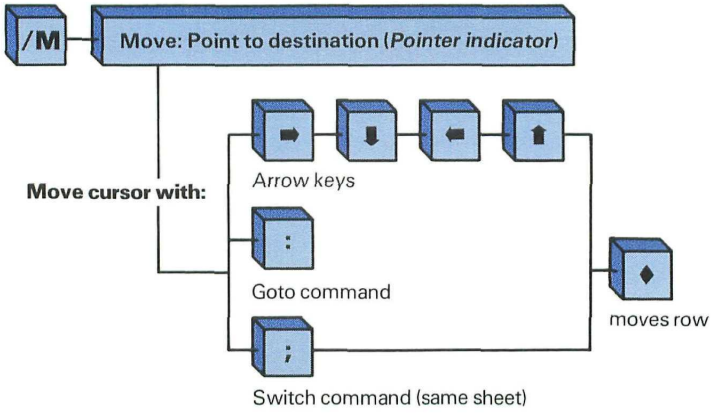
Related Commands:

- The Action command (!) performs the action appropriate to the current sheet, including model solution, list value fill-in, and plot and table display.
- The Solve command (/!) solves the currently defined model from any sheet.

/M Move Command

The Move command moves the current row to the specified destination.

/M Move row



Description:

Typing /M*pointing* ♦ moves the current row to the indicated final cursor position on the same sheet. The cursor may be positioned on any field in the row to be moved. The Move command copies the complete contents of the fields on a row, not only the portions of items displayed in the fields. The Move command is completed by pressing the Enter key.

The program prompts for the destination for the move and indicates the starting cursor position on the Prompt/Error line:

`Move: Point to destination (`*Pointer Indicator*`)`

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position as it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting cursor position. The Arrow keys may be used in any combination in any direction with the Goto and Switch commands to reach the final cursor position on the same sheet. When the Move command is completed by pressing the Enter key, the cursor remains at the destination, other rows are moved down to make space at the destination, or up to fill the space vacated by the move. The Move command prompt disappears.

Usage Notes:

- The Goto command (:) Destination or Search options may be used to move the cursor to the final destination. The Pointer Indicator is not displayed during the Goto subset of the Move command.
- The Switch command (;) may be used to switch windows so that a row can be easily moved from one view of a sheet to another. Both windows must contain the same sheet prior to beginning the Move command.
- A row can only be moved in the Field Area of a sheet.
- If a blank row is moved to another location, a blank row is inserted at the destination.

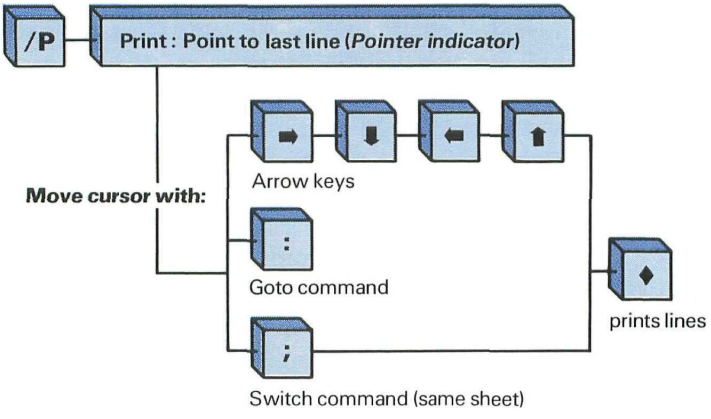
Related Command:

- The Copy command (/C) copies the current item to the specified destination.

/P Print Command

The Print command prints the specified rows from the current sheet onto an output device or into a file.

/P Print lines



Description:

Typing /P*pointing* ♦ prints the block of rows on the current sheet from the starting cursor position to the final cursor position. The cursor may be positioned on any field on the first row to be printed.

The program prompts for the last line to print and indicates the starting position on the Prompt/Error line:

`Print: Point to last line (`*Pointer Indicator*`)`

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position as it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting cursor position. The Arrow keys may be used in combination in any direction with the Goto and Switch commands to reach the final cursor position on the same sheet. When the Print command is completed by pressing the Enter key, the current screen is replaced with a display of the lines being printed.

The specified lines are printed on the printer or to the filename specified in the Printer Device or Filename field on the Global Sheet. (See the Hardware Reference section for valid filenames and printer device names for your computer.) The output is formatted according to the options specified in the other Printer fields on the Global Sheet and displayed simultaneously on the screen. If the Printer Device or Filename field is blank, the lines are displayed only to the screen and it is necessary to press the Enter key to display each page after the first, and then, after the last page, return to the previous display.

Pressing the Break key interrupts the printing action and returns immediately to the previous display.

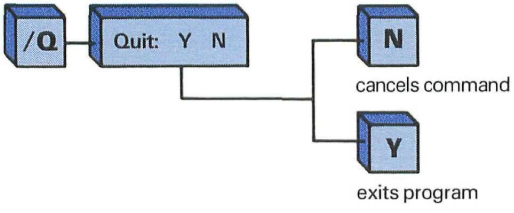
Usage Notes:

- The Goto command (:) Destination or Search options may be used to move the cursor towards the last row to print. The Pointer Indicator is not displayed during the Goto subset of the Print command.
- The Switch command (;) may be used to switch the cursor from one view of a sheet in one window to another view of the same sheet in the other window. Both windows must contain the same sheet prior to beginning the Print command. This feature is useful in printing long sheets.
- The block of rows to be printed can be from both the Header Area and the Field Area.

/Q Quit Command

The Quit command exits the TK!Solver program.

/Q Quit program



Description:
Typing /**QY** exits the TK!Solver program.

Typing /**QN** or /**Q**⊗ cancels the Quit command. The computer produces an error signal, the Quit prompt disappears from the Prompt/Error line, and the user remains in the program.

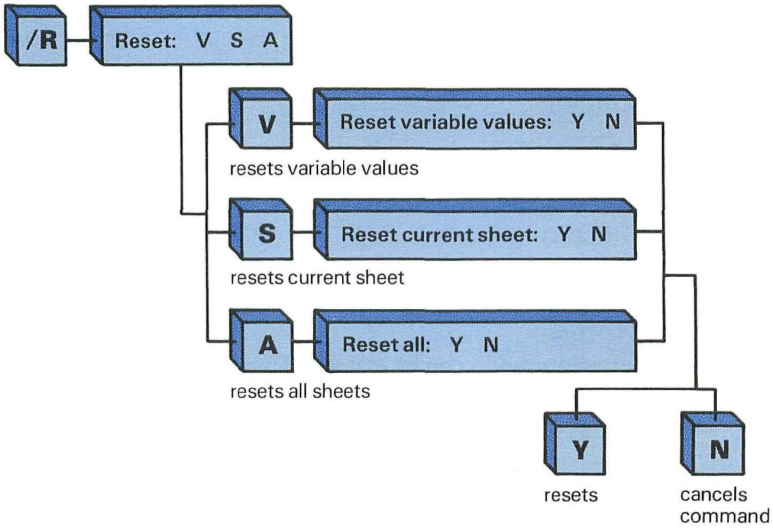
Usage Notes:
- The program can only be exited from Command mode.
- All current data and user settings are lost unless saved with the Storage command (/S) before quitting.
- Using the Reset command (/RAY) is equivalent to exiting the program (/QY) and restarting it.

Related Commands:
- The Storage command (/S) saves and reloads files containing models or portions of models.
- The Reset command (/R) clears data from sheets or portions of sheets.

/R Reset Command

The Reset command clears variable values, the current sheet, or all data from the program.

/R Reset program, sheet, or values



Description:

Variable Option (/RV)

Typing the Variable option (/RV) clears all values from the Input and Output fields and any **G** settings in the Status field on the Variable Sheet and all of its subsheets. The cursor may be positioned on any sheet. The program prompts for confirmation:

```
Reset variable values: Y N
```

Typing **Y** clears variable values; typing **N** cancels the Reset command. A Status message is displayed while the program clears data: "Resetting variable values".

Sheet Option (/RS)

Typing the Sheet option (/RS) clears all field items and resets any program defaults on the current sheet; the cursor must be positioned on the sheet to be reset. The cursor is moved to the initial cursor position for that sheet. The program prompts for confirmation:

Reset

```
Reset current sheet: Y N
```

Typing **Y** resets the current sheet; typing **N** cancels the Reset command. A Status message is displayed while the program clears data: "Resetting current sheet".

All Option (/RA)
Typing the All option (/RA) clears all field items on all sheets and resets program default values. It is equivalent to re-initializing the program. The cursor may be positioned on any sheet. The program prompts for confirmation:

```
Reset all: Y N
```

Typing **Y** resets the program; typing **N** cancels the Reset command. A Status message is displayed while the program clears data: "Resetting everything".

Usage Notes:

- The Reset command can be cancelled by pressing the Break key any time before the final **Y** confirmation. Once **Y** has been typed the resetting action cannot be interrupted.

Related Commands:

- The Storage command (/S) saves and reloads files containing models or portions of models.
- The Quit command (/Q) exits the TK!Solver program and clears all data.

< Return Command

The Return command returns from the display of a subsheet to the previous sheet or subsheet.

< Return



returns from subsheet

Description:
Typing 〈 from a subsheet returns the display of the previous sheet, including the previous cursor position, or subsheet in the same window.

It is possible to return from a List subsheet through its corresponding Variable subsheet to the Variable Sheet, or from a domain or range list subsheet to the User Function Sheet, by using the Return command twice in succession.

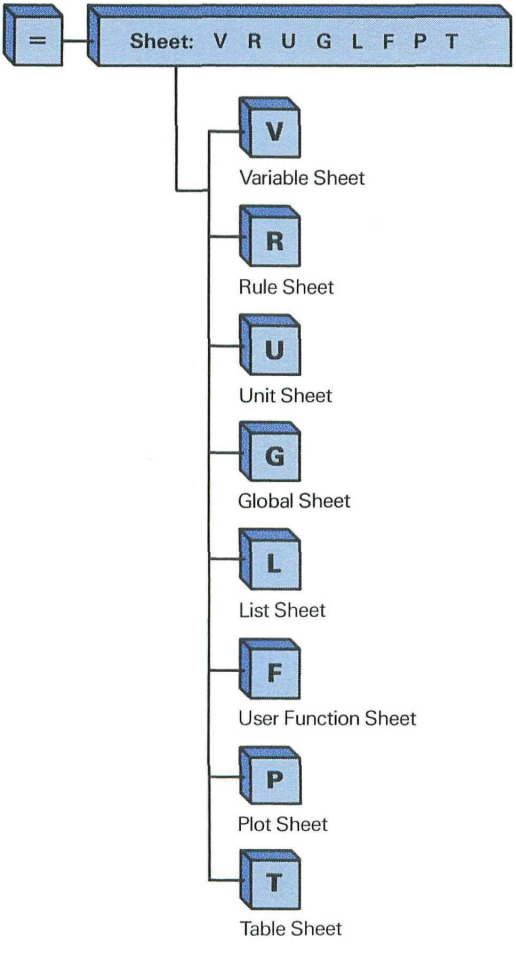
Usage Notes:
- The Select command (=) can be used, instead of the Return command, to return from a subsheet directly to another specified sheet.

Related Commands:
- The Dive command (〉) displays a dependent subsheet for the current sheet or subsheet in the same window.
- The Select command (=) displays the specified sheet in the current window.

= Select Command

The Select command displays the specified sheet in the current window.

= Select sheet



Description:

Typing =*sheet letter* displays the specified sheet in the current window, causing the previous sheet to disappear:

V	Selects the Variable Sheet
R	Selects the Rule Sheet
U	Selects the Unit Sheet
G	Selects the Global Sheet
L	Selects the List Sheet
F	Selects the User Function Sheet
P	Selects the Plot Sheet
T	Selects the Table Sheet

If it is the first time the specified sheet has been selected during the current session, the cursor moves to the initial cursor position for the selected sheet. If the specified sheet is being re-accessed, the cursor moves to the most recent display and cursor position for that sheet.

Usage Notes:

- The Select command may be used to access any sheet from a subsheet without using the Return command (〈).

Errors

- Typing = followed by ⊗ or any character which is not a sheet designator cancels the Select command. The computer produces an error signal, the Select prompt disappears from the Prompt/Error line, and the command is cancelled.

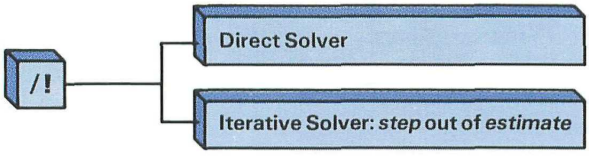
Related Commands:

- The Two-window option of the Window command (/W*sheet letter*) splits the screen into two windows from a one-window display, with the current sheet in the top window and the specified sheet in the bottom window. The One-window option (/W1) displays a single window in the Information Area, retaining only the current sheet from a two-window display.
- The Switch command (;) moves the cursor from the current window to the other window.
- The Dive command (〉) displays a dependent subsheet for the current sheet or subsheet in the same window.
- The Return command (〈) returns the display of the previous sheet or subsheet to the same window after a Dive command (〉).

/! Solve Command

The Solve command solves the currently defined model from any sheet.

/! Solve model



Description:
Typing /! with the cursor positioned in any field on any sheet solves the currently defined model by invoking the Direct Solver or the Iterative Solver. A Status message appears on the Prompt/Error line indicating the solver being used:

```
Direct Solver
```

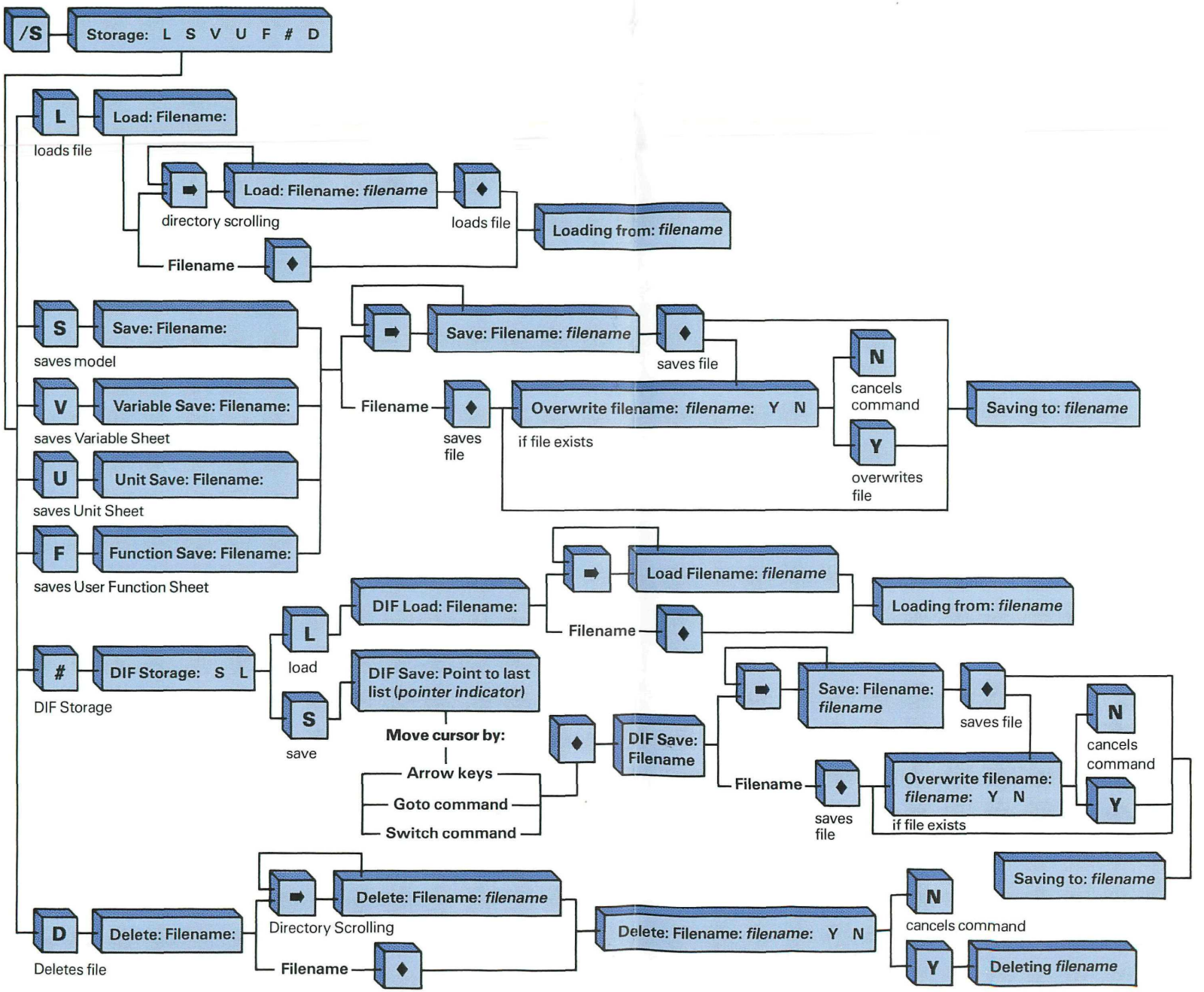
or

```
Iterative Solver: step out of estimate
```

For further discussion of the solvers, see the Solve action of the Action command (!) and the Mathematics section.

Related Commands:
- The Solve action of the Action command (!) solves the currently defined model from the Variable or Rule Sheet only.
- The Solve option (/L!) and Block option (/LB) of the List command solve a model for lists of values associated with variables.

/S Storage


/S
Storage: L S V U F # D
L
Load: Filename:
loads file
directory scrolling
Load: Filename: filename
loads file
Loading from: filename
Filename
S
Save: Filename:
saves model
Save: Filename: filename
saves file
V
Variable Save: Filename:
saves Variable Sheet
Filename
saves file
Overwrite filename: filename: Y N
if file exists
N
cancels command
Y
overwrites file
Saving to: filename
U
Unit Save: Filename:
saves Unit Sheet
F
Function Save: Filename:
saves User Function Sheet
#
DIF Storage: S L
DIF Storage
L
load
DIF Load: Filename:
Load Filename: filename
Filename
Loading from: filename
S
save
DIF Save: Point to last list (pointer indicator)
Move cursor by:
Arrow keys
Goto command
Switch command
DIF Save: Filename
Save: Filename: filename
saves file
Filename
saves file
Overwrite filename: filename: Y N
if file exists
N
cancels command
Y
Saving to: filename
D
Delete: Filename:
Deletes file
Directory Scrolling
Delete: Filename: filename
Filename
Delete: Filename: filename: Y N
N
cancels command
Y
Deleting filename

/S Storage Command

The Storage command saves, loads, or deletes files containing models or portions of models.

Description:
Typing /S calls the storage facility and offers a choice of options for loading, saving, or deleting files:

```
Storage: L S V U F # D
```

Each option of the Storage command prompts for a filename. (See the Hardware Reference section for information on valid file specifications for your computer.)

A new filename may be specified, with a disk drive specifier if one is necessary, and completed with the Enter key, or Directory Scrolling may be used to select an existing filename from the filenames already on an inserted diskette.

Directory Scrolling selects appropriate files from the default or specified diskette according to the option of the Storage command:

- the Delete file option of the Storage command (/SD) selects all files on the diskette;
- the DIF Save (/S#S) and DIF Load (/S#L) options select only DIF files;
- all other options of the Storage command select only TK!Solver storage files.

If a filename is omitted or partially specified for any option of the Storage command, Directory Scrolling displays each matching filename so that one may be selected and used. Directory scrolling is initiated by pressing the Right Arrow key (➡) alone without any file specification, or after a partial file specification.

A partial file specification is determined by the computer system being used and might include a disk specifier by itself, a file specification using system wildcards, a filename without an extension (for the Delete file option), or some combination with a disk specifier. (See the Hardware Reference section for more information on valid file and disk specifications and the use of wildcards and file extensions, if any, for your system.)

Directory Scrolling displays the appropriate prompt and the first selected filename on the Prompt/Error line:

```
Load filename: filename
Save filename: filename
Delete filename: filename
```

Filenames can be scrolled through in only one direction by pressing the Right Arrow key. The current filename may be selected by pressing the Enter key (♦). If all the files are scrolled through before the Enter key is pressed, the Storage command is cancelled.

Typing /S# calls the DIF storage facility and offers two options for saving and loading DIF format files:

```
DIF Storage: S L
```

DIF files contain lists of data in a format interchangeable with other programs. All labels and string values in a DIF file must conform to the TK!Solver conventions for valid name items and symbolic values before the file can be loaded into the TK!Solver program correctly. (For more information about DIF files, see the appendix on TK!Solver Files.)

Saving:

If a filename already exists, the program prompts for overwriting:

```
Save: Overwrite filename: filename: Y N
```

Typing **Y** copies the new data to the specified filename and erases any previous data; typing **N** cancels the Storage command.

Save Model Option (/SS)
Typing /**SS** displays a prompt for a filename:

```
Save: Filename:
```

Typing /SS*filename*♦ saves current data from all sheets to the specified file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Saving to *filename*".

Variable Save Option (/SV)
Typing /**SV** displays a prompt for a filename:

```
Variable Save: Filename:
```

Typing /**SV***filename*♦ saves the current data from the Variable Sheet and Variable subsheets to the specified file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Saving to *filename*".

Unit Save Option (/SU)
Typing /**SU** displays a prompt for a filename:

```
Unit Save: Filename:
```

Typing /**SU***filename*♦ saves current data from the Unit Sheet to the specified file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Saving to *filename*".

User Function Save Option (/SF)
Typing /**SF** displays a prompt for a filename:

```
Function Save: Filename:
```

Typing /**SF***filename*♦ saves current data from the User Function Sheet, List Sheet, and related subsheets to the specified file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Saving to *filename*".

DIF Save Option (/S#S)
The DIF Save option can only be used from the List Sheet. The cursor must be positioned on the first list to be saved. Typing /**S#S** displays a prompt for the last list to save:

```
DIF Save: Point to last list (Pointer Indicator)
```

The Pointer Indicator is displayed at the end of the Prompt/Error line to indicate the current cursor position as it changes during pointing; the Position Indicator in the upper lefthand corner of the screen continues to display the starting cursor position. The Arrow keys may be used in any combination in any direction with the Goto and Switch commands to reach the final cursor position on the same sheet.

When the last list is indicated by pressing the Enter key, the prompt on the Prompt/Error line is replaced by a prompt for a filename:

```
DIF Save: Filename:
```

Typing /**S#S***pointing* ♦ *filename* ♦ saves the specified lists on the List Sheet in the specified DIF-format file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Saving to *filename*".

DIF files save information from the List Sheet and List subsheets, including Storage and Display units.

Loading:

Load Option (/SL)
Typing /**SL** displays a prompt for a filename:

```
Load: Filename:
```

Typing /**SL***filename* ♦ loads previously saved data from the specified file into the program. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Loading from *filename*".

When a file is loaded over a model already on the screen, the information from the file is added to what is already displayed. Information from the specified file from the Rule, Unit, Table, and Plot sheets is inserted at the bottom of the appropriate sheet, even if the same information then appears twice. Information from the Variable, List, and Function sheets is added to the bottom of the appropriate sheet only if there are no duplicate names. If names are duplicated, the values from the loaded file are assigned to the variable or user function names on the screen, overwriting any displayed values. If list names are duplicated, a new list is created without a name. Header Areas of sheets and subsheets are always reset by the loaded information.

DIF Load Option (/S#L)
Typing **/S#L** displays a prompt for a filename:

```
DIF Load: Filename:
```

Typing **/S#L***filename*♦ from the List Sheet loads the specified DIF-format file. Directory Scrolling can also be used. After the final Enter key is pressed, a Status message appears on the Prompt/Error line: "Loading from *filename*".

When a DIF file is loaded over a model already on the screen, the information from the file is added to what is already displayed on the List Sheet and subsheets. If names are duplicated, the list information from the DIF file is loaded without names below the values displayed on the screen.

Deleting:

Delete File Option (/SD)
Typing **/SD** displays a prompt for a filename:

```
Delete: Filename:
```

Typing **/SD***filename*♦ deletes the specified file from the specified or current diskette. Directory Scrolling can also be used. If you specify a filename without an extension, the program finds the first instance of the filename with any extension and prompts for confirmation of the complete filename. ♦ selects that filename. ➡ scrolls to the next instance of the filename.

After the final Enter key is pressed, the program prompts for confirmation of the delete.

```
Delete filename: filename: Y N
```

Typing **Y** deletes the current file; typing **N** cancels the Storage command. After the **Y** confirmation is typed, a Status message appears on the Prompt/Error line: "Deleting *filename*".

Usage Notes:

- The Edit command (⊡) can be used to edit the filename on the Prompt/Error line any time before the Enter key is pressed to complete the filename or the Right Arrow key is pressed to begin Directory Scrolling.

Related Commands:

- The Reset command (/R) clears the program or data from sheets.
- The Quit command (/Q) exits the program. The Storage command must be used to save data since the Quit command clears the program.

; Switch Command

The Switch command moves the cursor between windows.

; Switch window



moves cursor between windows

Description:
Typing the Switch command (;) with two windows displayed on the screen moves the cursor from its present position to the other window. The cursor is moved to the initial cursor position or the most recent cursor position for the sheet or subsheet in that window. Typed a second time, the Switch command returns the cursor to its position on the original window.

Usage Notes:
- The Switch command can be used with the Copy (/C) command to move towards a final destination on any sheet displayed in the other window.
- The Switch command can be used to move towards the final row or destination for the Blank (/B), Delete (/D), Move (/M), Print (/P), and DIF Save (/S#S) commands if both windows contain views of the same sheet.

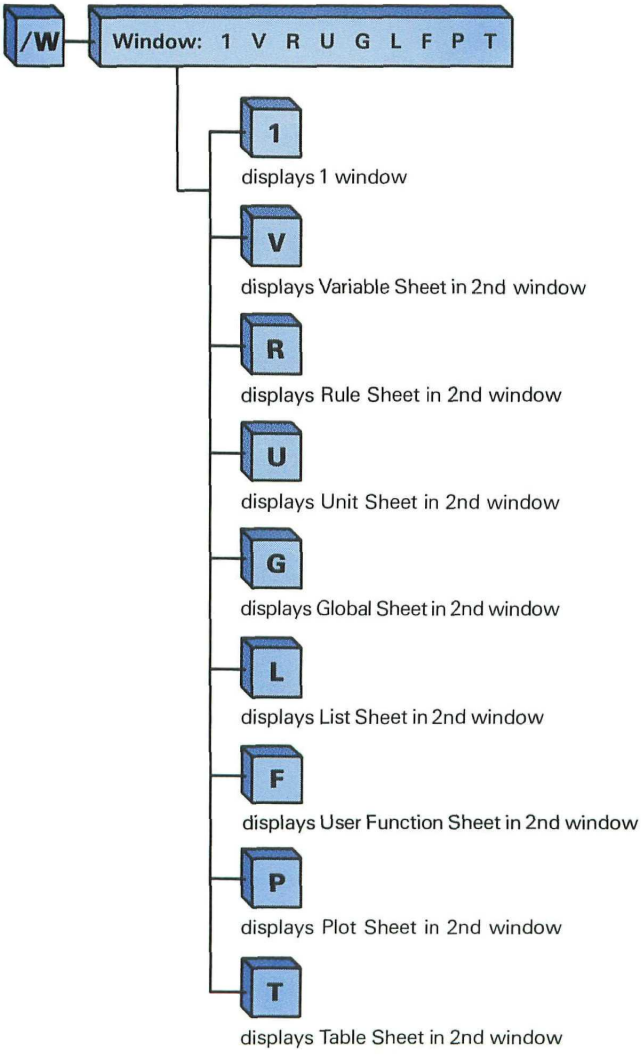
Errors:
- If the Switch command is typed and one window is displayed, the computer produces an error signal. The cursor does not move.

Related Commands:
- The Two-window option of the Window command (/W*sheet letter*) splits the screen into two windows when only one window is displayed on the screen.
- The Select command (=) displays the specified sheet in the current window.

/W Window Command

The Window command displays a single window or splits the screen into two windows.

/W Display window



Description:

One-window Option (/W1)

Typing the One-window option (/W1) when two windows are displayed creates one window in the Information Area. To retain a sheet, type **/W1** with the cursor on any field on that sheet. The sheet expands to fill the Information Area, and the other sheet disappears. The cursor remains on the same field.

Two-window Option (/W*sheet letter*)

Typing the Two-window option (/W*sheet letter*) when one window is displayed splits the Information Area into two windows.

To split a single window into two equal-sized windows, type **/W***sheet letter*:

V Displays the Variable Sheet
R Displays the Rule Sheet
U Displays the Unit Sheet
G Displays the Global Sheet
F Displays the User Function Sheet
L Displays the List Sheet
P Displays the Plot Sheet
T Displays the Table Sheet

The upper portion of the current sheet remains in the top window, the specified sheet displays in the newly created bottom window, causing the lower portion of the current sheet to disappear. The cursor moves to the initial cursor position for the specified sheet in the bottom window.

Usage Notes:

- When the TK!Solver program is loaded, two windows are displayed. The top window contains the Variable Sheet and the bottom one the Rule Sheet. The initial cursor position is in the Rule field on the first row of the Rule Sheet.
- Two different views of the same sheet can be displayed simultaneously in two windows. Changes made to one display of a sheet appear on the other.
- The Window command can be used from a subsheet as well as from a sheet.
- The Window command does not function during actions that pre-empt the entire screen, such as use of the HELP Facility (?), the Plot and Table actions (!), or the Print command (/P).

Errors:
Certain errors cancel the Window command. The computer produces an error signal, the Window prompt disappears from the Prompt/Error line, and the command is cancelled. Errors occur when:

- **/W1** is typed and there is one window displayed on the screen;
- **/W***sheet letter* is typed and there are two windows displayed on the screen;
- **/W** is followed by ⊗ or any character which is not an option of the Window command.

Related Commands:
- The Switch command (;) moves the cursor from the current window to the other window.
- The Select command (=) displays the specified sheet in the current window.
- The Dive command (›) displays a dependent subsheet for the current sheet or subsheet in the current window.
- The Return command (‹) returns from the display of a subsheet to the previous sheet or subsheet in the same window after a Dive command (›).

Mathematics

Equation Components

This section details the components of rules or equations in the TK!Solver program. In the TK!Solver program, equations are written in standard algebraic notation. Standard computer conventions are used to represent arithmetic operators. These conventions are detailed in the Arithmetic Operators section below.

Values in the TK!Solver program can be either numeric values or symbolic values.

Numeric Values

Use
Specify quantity.

Format
Decimal Form and Scientific Notation

☐ Decimal Form
Real numbers written in standard form.

Examples:

12.5	-6.7893
321313243125	.54783

The TK!Solver program does not accept commas in numeric input. For numbers containing a decimal point, a leading "0" entered to the left of the decimal point will be deleted by the program.

☐ Scientific Notation
Real numbers written as a product of a number whose absolute value is between 1 and 10 and a power of ten. The TK!Solver syntax for scientific notation is the number between 1 and 10 separated from the exponent of 10 by an "E".

Examples:

$3.4 \times 10^{4} = 3.4E4$

$5.676 \times 10^{-3} = 5.676E-3$

The "E" can be entered in either lowercase or uppercase, but is always displayed as uppercase. Entries may be typed with numbers less than 1 or greater than 10 before the "E", but all output is in standard form.

Range of Numbers
System dependent. See the Hardware Reference section for the range for your system.

Precision
All values are rounded to fit display fields, without affecting the internal representation of the number. The maximum number of significant digits is 11–12. The full precision of any displayed number can be seen by moving the cursor over the number and reading the display on the Status line. Digits beyond the significant digits are displayed as zeros.

Symbolic Values

Use
Represent a non-numeric value.

Format
Any alphanumeric set of characters including the special characters:

@ # $ % _

Symbolic values cannot contain spaces or begin with a number; they must be preceded by an apostrophe to be entered in the program. These values are case sensitive on computers that support both cases.

Examples:

'a	'B2	'b2	'%tj3
'wk_4@%	'Q756	'$h6H5	

Maximum Length: 200 characters

Variables

Use

A symbol used to represent any element of a set of values.

Format

Must begin with an alphabetic character or one of the special characters shown below and contain only alphanumeric characters and the special characters:

@ # $ % _

They cannot contain spaces. Variables in the TK!Solver program are case sensitive on computers that support both cases.

Maximum Length: 200 characters

Arithmetic Operators

Use

Designate appropriate computation.

Operators

The following symbols are used to specify the standard arithmetic operators:

+	Addition
-	Subtraction
*	Multiplication
/	Division
^	Exponentiation

All multiplication must be indicated by an asterisk (*); there is no implicit multiplication in the program.

Example:

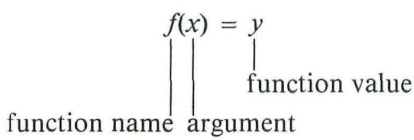
$4(x-y)(3x+y)$
must be written as
$4*(x-y)*(3*x+y)$

For precedence, see section on Evaluation of Expressions below.

Functions

There are two types of functions in the TK!Solver program, built-in functions and user functions.

Syntax



Function Name

☐ Built-in Functions

The names of the built-in functions appear in the next part of the manual. Names of built-in functions are case insensitive.

☐ User Functions

Must begin with a letter and contain only alphanumeric characters and the special characters:

@ # $ % _

Argument

Any valid TK!Solver expression, including symbolic values. Some built-in functions take multiple arguments.

Function value

Any valid TK!Solver value, including symbolic values.

Expressions

TK!Solver expressions have the same form as algebraic expressions.

Examples:

$2+x$ $\quad\quad$ $3*\sin(3*\text{pi}())/2$

$4*x\text{^}3-2*\text{int}(a/b)\text{^}(-c+d)$

Maximum Length: 200 characters

Rules

In the TK!Solver program all rules are equations. Equations are made up of two expressions joined by an equal sign.

Examples:
a/*b*^3–4∗*b* = sqrt(*a*∗*b*)–3E12
time = *period*/(1 + .25∗sin(*theta*/4)^2)
s = *v*∗*t* + 1/2∗*a*∗*t*^2

Maximum Length: 200 characters

Expression Evaluation

This section describes the rules of precedence and how the calculator feature of the program works.

Precedence

Within an expression, the order of operations performed is:

+	-	Unary plus or minus
^		Exponentiation
/	*	Division or multiplication
+	-	Binary addition or subtraction

Operations at the same level are performed left to right. Precedence can be altered by the use of parentheses. Operations inside parentheses are performed before any others.

Example:

-5 + 33/(5*3-4) - 2^3/1.6

Step-by-step solution:
-5 + 33/(5*3-4) - 2^3/1.6
-5 + 33/(15-4) - 2^3/1.6
-5 + 33/11 - 2^3/1.6
-5 + 33/11 - 8/1.6
-5 + 3 - 8/1.6
-5 + 3 - 5
-2 - 5
-7

Evaluation

Expressions are evaluated according to the rules of precedence. Variables are replaced with their given or computed value and numeric and symbolic values evaluate to themselves.

The Calculator

Any field that accepts numeric entries can be used as a calculator to evaluate expressions. Fields that permit only integer values accept only expressions that evaluate to integers. If a variable name is used as part of an expression to be evaluated, the program uses the value of the Calculation unit, not the Display unit. When values are displayed in mixed units, make cautious use of the Calculator.

Model Solution

This section begins with the solution process for a single equation. The remainder of the section describes in detail the Direct and Iterative Solvers of the program.

Direct Solver

Use

Solves models. It is activated by typing /! on any sheet or ! on the Variable or Rule Sheet.

Prompt/Error Line Message

```
Direct Solver
```

Single-Rule Evaluation

Evaluation of a model containing a single equation is very similiar to the evaluation of a model containing many equations. The expression on each side of the equal sign is evaluated separately. If each can be evaluated completely, the comparison tolerance is used to check the equality of the two expressions. If they are not equal, an appropriate error message is generated.

If the two expressions cannot be evaluated, the program attempts to algebraically place the unknown value on one side of the equation. Once this is accomplished, the remaining expression is evaluated to provide the value of the unknown. If this algebraic manipulation fails to solve the problem, an Iterative solution may still be possible.

In certain cases the value of a variable is indeterminable. In six of these cases, the TK!Solver program marks the rule either satisfied or unsatisfied without further evaluation of the rule. These cases are:

ATAN2(0,0) = *x*	Since ATAN is undefined at this point, the program returns no value and leaves the rule unsatisfied.
1 = 1^*x*	Since *x* can be any value, the program returns no value, yet marks the rule satisfied.
1 = *x*^0	Since *x* can be any non-zero value, the program returns no value, yet marks the rule satisfied.

$0=0\text{^}x$	Since 0 to any non-zero power must equal 0, the program returns no value, yet marks the rule satisfied.
$x=0\text{^}0$	Since $0\text{^}0$ is undefined, the program returns no value and leaves the rule unsatisfied.
$x=0/0$	Since $0/0$ is undefined, the program returns no value and leaves the rule unsatisfied.

In order for the Direct Solver to solve a single equation, it must be possible for it to isolate the variable being solved for. Three conditions must be met to make this possible:

- All variables in the equation, except the one being solved for, must have input values.
- The variable being solved for must appear only once in the equation.
 For example, the Direct Solver CANNOT solve for x in these equations:

 $x+x=y$ (can solve written as $2*x=y$)
 $x*x=y$ (can solve written as $x\text{^}2=y$)

- The variable being solved for cannot be the argument of a function without a unique inverse.
 For example, the Direct Solver CANNOT solve for x in these equations:

 SGN(x) = y INT(x) = y ABS(x) = y

Model Solution Process

The Direct Solver attempts to evaluate every equation in a model. As it satisfies the equality conditions for one equation, it uses the values it determined in that solution process to satisfy other equations.

As the Direct Solver determines unknown values it not only checks that the values are consistent with the equation involved, but with all other known values in the model.

Comparison Tolerance

The comparison tolerance is used by the solvers of the program to determine equality. Two values, x and y, are considered equal if the following inequality is true:

$$\frac{|x - y|}{\max(|x|,|y|)} \leq \text{comparison tolerance}$$

For values near zero, determined by the typical value, the comparison tolerance formula for equality changes to:

$$\frac{|x - y|}{\text{typical value}} \leq \text{comparison tolerance}$$

As the comparison tolerance gets smaller, the solution process becomes accurate to more decimal places.

Default value: .000001 (set on Global Sheet)

Typical Value

The typical value is used in conjunction with the comparison tolerance. The product of the typical value and the comparison tolerance defines at what point the program will treat values as near zero. For example, if:

Comparison tolerance = .000001
Typical value = .01

Then, any number with an absolute value less than (.000001)*(.01) = .00000001, or 1E–8, is considered near zero.

When the absolute value of a quantity approachs zero as defined by the above product, the ratio associated with the comparison tolerance changes. The change replaces "max(x,y)" with "typical value" for the denominator of the ratio.

Since typical value alters at what decimal place zero equivalence is obtained, it can be a useful tool. Increases in the typical value will have a concurrent increase in the decimal place at which values are considered equal to zero and vice versa.

Default value: 1 (set on Global Sheet)

Iterative Solver

Use

Solves models that the Direct Solver is unable to solve. To activate the Iterative Solver follow these steps:

- Determine which variables cannot be solved by the Direct Solver: Either check the conditions for Direct solution or perform a Direct solution.
- Type **G** in the Status field of each variable that requires an Iterative solution.
- Enter a guess value in the Input field of each variable requiring an Iterative solution.
- Type **/!** on any sheet or **!** on the Variable or Rule Sheets.

Prompt/Error Line Message

`Iterative Solver:` *step* `out of` *estimate*

The "step" in this message relates what iteration step the solver is currently performing. The "estimate" is an estimate of the total number of steps the Iterative Solver will need to solve the problem.

This number changes frequently during solving. If it stays higher than 20, this may indicate either a poor initial guess or possibly a problem without a solution.

Solution Process

The Iterative Solver uses the initial guess value(s) to solve the model. The equation(s) containing the iterated variable(s) and the entire model are checked for equality by the comparison tolerance. If there are any inconsistencies, another guess is generated by the solver. The comparison checks are performed on each step of the iteration. The Iterative Solver stops when two successive guesses are evaluated as equal and all equations possible in the model are satisfied.

Intermediate Redisplay

Each step of iteration generates a trial solution of the model. These values can either be displayed or suppressed. The mechanism for control of display is the Intermediate Redisplay ON field on the Global Sheet. If this field is set to "Yes", the values for each step of iteration are displayed. Set to "No", the values

are suppressed. Suppression of the intermediate values will shorten processing time for the Iterative Solver. The default value for this field is "Yes".

Maximum Iteration Count
The number of iteration steps that are performed by the Iterative Solver is set on Global Sheet in the Maximum Iteration Count field. The default value is "10". If the model does not solve in the number of steps set, typing the Action command again provides for another full set of iterations.

Guess Values
A list solving can help find an appropriate guess value for an Iterative solution of a polynomial. Set up a wide range of values for the variable which requires Iterative solution. The output list of polynomial values can point to possible guess values. A root of the equation will be between any two values of the polynomial with different signs.

Example:

$x^8 - 5x^6 + 3x^2 + 10x + 1 = y$

The roots of this equation are not obvious. Applying the method described above, make a list of 100 elements for x ranging from 20 to -20. A List solving for y shows four changes of sign. Any one of these can be used as a guess value for x with y set to "0".

To determine multiple roots of a polynomial, division of the polynomial by terms of the form $(x-r)$ where r is a root of the polynomial will result in a different root being found on the next attempted solution.

Comparison Tolerance and Typical Value
See under Direct Solver.

Automatic Iteration

Use
If a model requires iteration for particular sets of input values, Automatic Iteration alleviates the need to explicitly enter a guess value for variables for each solution of the model. It is particularly useful in conjunction with the Solve option of the List Command.

To use Automatic Iteration:

- Set the Automatic Iteration ON field on the Global Sheet to "Yes", the default value.
- On the Variable subsheet of each variable that may require an Iterative solution provide an initial guess value in the First Guess field.
- Type ! or /! to solve the model.

Solution Process
The Direct Solver calls the Iterative Solver if it is unable to solve the model and the appropriate Automatic Iteration values are set.

For some models it is difficult to set a guess value general enough to be useful for all input values of interest. Automatic Iteration is best suited for models that will be solved repeatedly within a moderate range of values.

Model Development Overview

TK!Solver models define relationships among variables within a system of equations. A model may be as simple as a single equation or so complex that it involves all the sheets and has over 100 equations.

Simple Models

The simplest model possible in the TK!Solver program involves only the Variable and Rule sheets and contains a single equation such as either of the following:

$a - c = b * d$
Volume = *length***height***depth*

The Rule Sheet contains the equation and the Variable Sheet, the variables.

Model Expansion Overview

There are several ways to expand a simple model. The options include the following enhancements:

- Entering new rules that refine or expand the model's utility
- Using TK!Solver functions
- Associating units with variables
- Associating lists of values with variables
- Creating functions to relate values
- Documenting the model with comments

Model Development Using Sheets

In this section each sheet is listed with a number of ways to use the sheet when building complex models. The lists of ideas are not exhaustive.

Variable Sheet

Units
Write equations in the units most convenient for calculation, with whatever units are appropriate chosen for the display of the variable on the Variable Sheet.

Lists
Associate lists with variables to solve models for several values of selected variables.

Iterative Solver
Solve models for variables not found by the Direct Solver.

Comments
Document variables.

Rule Sheet

List Functions
Several TK!Solver built-in functions manipulate lists. They are COUNT, DOT, ELEMENT, MAX, MIN, and SUM. Rules may be created to use these functions.

Editing Rules
Rules may be edited as long as the expressions on both sides of the equal sign remain valid and the overall character length does not exceed 200.

Comments
Document rules for further reference.

Unit Sheet

Variables
Provide unit conversions from Calculation to Display units. Also permits unit conversions among variables.

Lists
Provide unit conversions between the Storage units of lists and the Calculation units of variables.

Global Sheet

Solving
Speed processing time for Iterative and List solutions by switching Intermediate Redisplay ON to "No".

Automatic Iteration
Set models to switch to Iterative Solver when the Direct Solver is unable to fully solve a model. Automatic Iteration ON must be set to "Yes" and appropriate first guesses set on selected Variable subsheets.

Precision
See Comparison Tolerance and Typical Value for details affecting equality and zero determination using these features.

Printing
Set parameters affecting printing.

List Sheet and List Subsheets

Lists
Create lists to be used with the Solve option of the List command.

Functions
Create lists to be used with functions.

User Function Sheet

Functions
Create functions for use in equations.

Plot Sheet

Iterative Solver
If looking for simultaneous solutions to a set of equations, solve the model for a list of wide-ranging values and then plot the function values to determine the appropriate neighborhood for guess values.

Both Solvers
After list solving, plot to determine behavior of data.

Table Sheet

Lists
Display lists either for purposes of analysis or to see a table of results.

Model Development Using TK!Solver Features

In this section, three features of the TK!Solver program (units, lists, and functions) are described in terms of how they can be used in developing models.

Units

Although units are not necessary in models, when they are used, they affect every sheet, except the Global Sheet. This section contains three parts:

- On the Unit Sheet: Describes how to set up unit conversions on the Unit Sheet.
- On other Sheets: Describes each sheet on which units can be used.
- Between Sheets: Describes unit interactions with the sheets of the program.

On the Unit Sheet

There are four columns of fields on the Unit Sheet. The Multiply By fields relate the units in the From and To fields. The Add offset field adjusts the zeros of two related scales, such as Celsius and Fahrenheit temperature scales.

Examples:
yd and ft; Celsius and Fahrenheit

From	To	Multiply By	Add Offset
yd	ft	3	
C	F	1.8	32

Unit conversions can be made indirectly. An indirect conversion makes use of a third unit. In an indirect conversion, there must be a unit conversion between the third unit and each of the units to be converted.

Example:
yd and mi through ft

From	To	Multiply By	Add Offset
ft	yd	.3333333333	
mi	ft	5280	

Conversion from miles to yards is possible because each has a conversion defined in feet. It is not possible to extend this feature. Even if yd had a conversion defined in meters, the program could not convert mi to meters.

On Other Sheets

☐ Variable Sheet

Calculation Unit: The unit assumed for a variable when it is used in a rule.

Display Unit: The unit for display of variable value.

Unit Entry: The first entry in a unit field on this sheet sets the Calculation and Display units. Subsequent entries in the unit field change the Display unit, but do not affect the Calculation unit. Change in the Display unit is reflected on the Variable subsheet with the concurrent conversion of the value for the variable.

Unit Conversion: If the Display and Calculation units for any variable are different, their unit conversion must be defined on the Unit Sheet.

☐ Variable Subsheets

Calculation Unit: The unit assumed for a variable value when it is used in a rule.

Display Unit: The unit for display of a variable value.

Unit Entry: Calculation and Display values are entered on separate fields on this sheet. Change in the Display unit is reflected on the Variable Sheet with the concurent conversion of the value for the variable.

Unit Conversion: If the Display and Calculation units for any variable are different, their unit conversion must be defined on the Unit Sheet.

☐ List Sheet

Storage Unit: The unit of values in a list (DIF file) created in another program and/or the unit for conversion to the Calculation unit of an associated variable.

Display Unit: The unit for display of list values.

Unit Entry: The first entry in a Unit field on this sheet sets the Storage and Display units. Subsequent entries in the Unit field change the Display unit, but do not affect the Storage unit. Change in the Display unit is reflected on the List Subsheet with the concurrent conversion of the value for the variable.

Unit Conversion: If the Display and Storage units for any list are different, their unit conversion must be defined on the Unit Sheet.

☐ List Subsheets
Storage Unit: The unit of values in a list (DIF file) created in another program and/or the unit for conversion to the Calculation unit of an associated variable.

Display Unit: The unit for display of list values.

Unit Entry: Storage and Display values are entered on separate fields on this sheet. Change in the Display unit is reflected on the List subsheet with the concurrent conversion of the value for the variable.

Unit Conversion: If the Display and Storage units for any list are different, their unit conversion must be defined on the Unit Sheet.

Between Sheets
☐ Variable–Rule Sheets
Assignment of Display and Calculation units for variables can affect rules. All rule evaluation is performed in Calculation units. There must therefore be a coordination between equations containing variables associated with units and the assignment of units to the appropriate variables.

☐ Variable–List Sheets
For a list associated with a variable, the conversion from Calculation unit of the variable to Storage unit of the list must be defined on the Unit Sheet.

☐ User Function–List Sheets
If the lists involved in a user-defined function contain units, the function relationship is determined by the relationship between the Display units of the two lists.

If either list of the function is associated with a variable on the Variable Sheet, the Display unit of the list must be the same as the Calculation unit of the variable.

Lists

This section describes how lists are created and used.

List Creation

Lists are created on a List subsheet or a User Function subsheet. There are two methods for creating lists of values:

- Dive to a List subsheet or a User Function subsheet and type in each element of the desired list.
- For lists that are uniformly sequential and numerical, type the the first and last values in the appropriate element position. Typing **!** prompts:

  ```
  Fill List: Y N
  ```

 Typing **Y** fills in the intervening values using linear interpolation. Typing **N** cancels the command.

Unit Compatibility

Lists associated with units and used in the following ways need special attention when they are created:

- Associated with variables for list solution
- Used as domain and range lists for functions

Lists associated with variables must have a conversion provided on the Unit Sheet between the Storage unit of the list and the Calculation unit of the associated variable.

A list used as a domain or range for a function must be compatible with several units. If either list of the function is associated with a variable on the Variable Sheet, the Display unit of the list must be the same as the Calculation unit of the variable.

List Command

☐ Solve Option

The Solve option of the List command uses either the Direct or Iterative Solver to solve a model for a list or lists of input values creating the desired output list(s) of values.

The required steps for using the Solve option are:

- Determine which variables are to be associated with input and output lists.
- Type **L** in the Status field of each variable to be associated with a list.
- Assign a value to each variable associated with an input list. If no value is assigned, the variable is assumed to be output.
- Associate all input lists with the desired variables. Typing **L** in the Status field of a variable fills in the Associated List field on the Variable subsheet with the same name as the variable. If a list by a different name is to be associated with the variable, dive to the Variable subsheet and enter the different list name.
- Type **/L!** to solve the model.

☐ Block Option

The Block option of the List command solves for an interval of elements as specified by the user. The steps for using the Block option are:

- Set up the model for the Solve option of the List command.
- Type **/LB**.
- Enter the element number of the first element in the interval in response to the prompt:

```
List Block: First element:
```

- Enter the element number of the last element in the interval in response to the prompt:

```
List Block: Last element:
```

The Solve option is invoked for the specified interval.

☐ Put List

The Put option of the List Command places the current value of any variable associated with a list in the specified element position of the associated list. The steps for its use are:

- Set up the model for Solve option of the List Command.
- Type /**LP**.
- In response to the prompt:

  ```
  Put to element:
  ```

 enter the element number the current values should be assigned to.

☐ Get List

The Get option of the List Command gets the specified element value of any variable associated with a list and enters it on the Variable Sheet. The steps for its use are:

- Set up the model for Solve option of the List Command.
- Type /**LG**.
- In response to the prompt:

  ```
  Get from element:
  ```

 enter the element number of the values desired to be the current values on the Variable Sheet.

Functions

This section describes both user functions and built-in functions in the TK!Solver program.

User Functions

Syntax

See functions under Equation components.

Function Name

See functions under Equation components.

Defining

Lists can be associated with one another through the User Function Sheet. The lists can either be created on the User Function Subsheet or be ones that already exist on the List Sheet. The associations are defined through the Mapping field on the User Function Sheet (see section on mappings which follows). Enter the list that supplies the values for the function argument as the domain and the list that supplies the values for the function value as the range.

Mappings/Interpolation

There are three options for the Mapping field: Table, Linear, and Step. The mapping defines the type of interpolation desired between explicit values of related lists. Upon an entry in the Name field, the default value, Table, is displayed in the Mapping field.

The three options are:

☐ Table

No interpolation between explicit values.

☐ Linear

Linear interpolation between explicit values. For example, consider two lists, *A* and *B*:

List *A*	List *B*
1	10
2	20
3	30

To find the value, say *x*, in list *B* associated with 2.5 in the list *A*, linear interpolation, a proportional method, is used to find that 25 is the correct list *B* value.

$$\frac{\text{List } A}{\frac{2.5 - 2}{3 - 2}} = \frac{\text{List } B}{\frac{x - 20}{30 - 20}}$$

☐ Step
Interpolation such that for a given set of domain values, there is a constant value in the range. Between two domain values, the function evaluates to the range value associated with the lower numbered domain element.

Example:

List *A*	List *B*
1	10
2	20
3	30

To find the value in list *B* associated with 2.5 in the list *A*, step interpolation checks which elements of list *A* are involved and returns the range value associated with element 2, 20.

To define standard linear or step functions, all list values must be entered in either ascending or descending order because all interpolation is done point to point. No internal ordering takes place.

Irregular Interpolations
If lists used for User Functions are not in either ascending or descending order, interpolation may not act as expected. TK!Solver interpolation operates point to point. Given a value that requires interpolation, it looks only for the first interval of two values that can contain the desired point.

Example:

List *A*	List *B*
3	30
5	70
2	20
4	40

Using linear interpolation, a value of 3.8 in the domain returns a value of 46 because the interpolation interval was 30 to 70, not 30 to 40 as might be expected.

Built-in Functions

There are two distinct classes of built-in functions, standard mathematical functions and special TK!Solver functions. The mathematical functions are written in their standard form and return their normal function values, whereas the syntax and behavior of the TK!Solver functions should be checked before use.

The elementary transcendental functions, such as sin(x), tan(x), e^x, and y^x are typically evaluated by the TK!Solver program to an accuracy of 10 significant digits.

However, as on many computer systems, evaluation of some functions for extreme argument values can lead to erroneous results. Thus, if the argument of the sin(x) is 10^n, the accuracy of the function value diminishes as n increases in size. At about 10^{10} the accuracy associated with the function value is about two significant digits. The best values for evaluation of the trigonometric functions are in the principal value region.

Although specified throughout the documentation in uppercase, built-in functions names are case insensitive.

Built-in Functions

The following built-in functions are available in the TK!Solver program. They are divided into standard mathematical functions and special TK!Solver functions.

Mathematical Functions		TK!Solver Functions	
ABS	EXP	APPLY	MIN
ACOS	LN	ATAN2	NPV
ACOSH	LOG	COUNT	PI
ASIN	MOD	DOT	POLY
ASINH	SIN	E	SGN
ATAN	SINH	ELEMENT	SQRT
ATANH	TAN	GIVEN	STEP
COS	TANH	INT	SUM
COSH		MAX	

The following definitions are used for function arguments in the descriptions below. If more than one argument of the same type appears in the same function, they are numbered sequentially.

x — Any numeric-valued expression.

series — A set of two or more expressions separated by commas.

listname — Any expression that evaluates to a symbolic value that is a list name. **Note:** A symbolic value **must** be preceded by an apostrophe (').

ABS — Absolute Value
ABS(*x*)
Function value: the magnitude of the specified argument.

ACOS — Arc Cosine
ACOS(*x*) $-1 \leq x \leq +1$
Function value: the inverse cosine between 0 and π, in radians, of the specified argument.

ACOSH — Arc Hyperbolic Cosine
ACOSH(*x*) $x \geq 1$
Function value: the inverse hyperbolic cosine of the specified argument.

APPLY

Apply
APPLY(*function*, *domain*)
function: any value that evaluates to a user function name written as a symbolic value.

domain: any expression that evaluates to a value in the domain of the specified function.

Function value: The first argument evaluates to a user function name and this function is then evaluated for the value of the second argument.

Example: Use APPLY to generate the square or cube of the numbers 1,2,3,4 as desired.

Apply makes use of two functions for this example:

square(a) = b cube(a) = c
square(2) = 4 cube(2) = 8

apply(*function*, *x*) yields:

apply('square,2) = square(2) = 4
apply('cube, 2) = cube(2) = 8

ASIN

Arc Sine
ASIN(x) $-1 \leq x \leq +1$
Function value: the inverse sine between $-\pi/2$ and $\pi/2$, in radians, of the specified argument.

ASINH

Arc Hyperbolic Sine
ASINH(x)
Function Value: the inverse hyperbolic sine of the specified argument.

ATAN

Arc Tangent
ATAN(x)
Function value: the inverse tangent between $-\pi/2$ and $\pi/2$, in radians, of the specified argument.

ATANH

Arc Hyperbolic Tangent
ATANH(x) $-1 \leq x \leq +1$
Function value: the inverse hyperbolic tangent of the specified argument.

ATAN2 — Arc Tangent

ATAN2(*x1*, *x2*)

Function value: ATAN(*x1*/*x2*). Note this function returns the correct values by quadrant. This function returns values from 0 to 2π, as opposed to atan which only returns values in the interval $-\pi/2$ to $\pi/2$.

Example: Use ATAN2 to find: atan(-.7071068/-.7071068).

ATAN2(-.7071068, -.7071068) = -2.356194 radians (or -135 degrees)

COS — Cosine

COS(*x*) *x* in radians

Function value: the cosine of the specified argument.

COSH — Hyperbolic Cosine

COSH(*x*)

Function value: the hyperbolic cosine of the specified argument.

COUNT — Count

COUNT(*series*)

COUNT(*listname*)

Function value: the number of elements in the specified list or series, or 0 if the specified listname contains no elements.

DOT — Dot Product

DOT(*listname1*, *listname2*)

DOT(*listname1*, *series*)

Lists used with this function may contain blank entries which are treated as zeros.

Function value: the dot product of the two specified lists. Each element of the first list is multiplied by its corresponding element in the second list and the products are summed.

E — Natural Log Base

E() no argument

Function value: an approximation of the value of *e*.

ELEMENT — Element

ELEMENT() no argument

Function value: the number of the element currently being processed by the List Solver, or 0 if the List Solver is not processing.

ELEMENT(*listname*, *element number*)

Function value: the value of the specified element in the specified list.

ELEMENT(*listname*, *element number*, *x*)

Function value: *x* is a default function value. Returns the value of the specified element in the specified list or the specified default value, if that element does not exist.

Example: Use ELEMENT to determine the value of the 5th and 6th elements of list *Q*.

Element #	List Q	ELEMENT('Q, 5) = 11
1	3	ELEMENT('Q, 6, 1) = 1
2	5	
3	7	
4	9	
5	11	

EXP — Exponentiation

EXP(*x*)

Function value: *e* raised to the power of the specified argument.

GIVEN — Given Input

GIVEN(*variables*, *x1*, *x2*)

variables: A series of expressions that evaluate to variable names written as symbolic values.

x1 is the value to return if all the variables have input values.

x2 is the value to return if any of the variables do not have input values.

Function value: *x1* if all of the variables have input values, *x2* if any of the variables do not have input values.

Examples: Use GIVEN with the variables *a* and *b*.

Example 1:

Input	Name	
	a	GIVEN('*a*,'*b*, 0,1) = 1
5	*b*	

Example 2:

Input	Name	
3	*a*	GIVEN('*a*,'*b*, 0,1) = 0
2	*b*	

INT — Integer Truncation
INT(*x*)
Function value: the integer portion of the specified argument; effectively rounds down for positive numbers and up for negative numbers.

LN — Natural Logarithm
LN(*x*) *x*>0
Function value: the base *e* logarithm of the specified argument.

LOG — Logarithm
LOG(*x*) *x*>0
Function value: the base 10 logarithm of the specified argument.

MAX — Maximum
MAX(*series*)
MAX(*listname*)
Function value: the greatest value within the specified series or list. Symbolic values are ignored.

MIN — Minimum

MIN(*series*)

MIN(*listname*)

Function value: the least value within the specified series or list. Symbolic values are ignored.

MOD — (MODULUS) Modulo

MOD(*x1*, *x2*)

Function value: For *x1* and *x2* greater than 0, the remainder of division of *x1* by *x2*, which in the general case is defined as the unique value of *r* that satisfies: $x1 = x2 * n + r$, where *n* is an integer and *r* is between 0 and *x2* (including 0 and not including *x2*).

NPV — Net Present Value

NPV(*x*, *series*)

NPV(*x*, *listname*)

x = the rate.

series = cashflow values.

listname = list of cashflow values.

Function value: the net present value at the specified rate using the specified series or list of cash flow values. The result is equal to:

$$\sum_{i=0}^{n} \text{list}[i + 1]\left(\frac{1}{1 + x}\right)^{i}$$

where *i* takes values from 0 to *n* and $n+1$ is the number of elements in the specified list. The rate is any expression that evaluates to a percent written as a decimal. The first cash flow is taken to be the cash flow at time zero.

Example: Use NPV to evaluate the cash flows of -1000, 600, 400, 300, 150, 10, and 10 with a rate of 12%.

NPV(.12,-1000,600,400,300,150,10,10) = 174.19

PI — π

PI() no argument

Function value: an approximation for the value of π.

POLY — Polynomial

POLY(*x*, *series*)
POLY(*x*, *listname*)
x = the variable value for the polynomial.
series = coefficients of the polynomial beginning with the coefficient of the variable with the greatest exponent. Zeros should be entered for any nonexistent standard terms.
listname = a list of coefficients.
Function value: the value of the polynomial for the specified variable using the specified series or list of coefficients. The result is equal to the sum of $n+1$ terms of a polynomial of the form:

$$\sum_{i=0}^{n} \text{list}[i+1]x^{n-i}$$

where i takes values from 0 to n and there are $n+1$ elements in the specified list.

Example:

Use POLY to evaluate

$6x^5-3x^3+7x^2-2x+8$ for $x=4$

POLY(4,6,0,-3,7,-2,8) = 6064

SGN — (SIGNUM) Sign

SGN(*x*)
Function value:

$x<0$	$x=0$	$x>0$
$\text{sgn}(x)=-1$	$\text{sgn}(x)=0$	$\text{sgn}(x)=1$

SIN — Sine

SIN(*x*) *x* in radians
Function value: the sine of the specified argument.

SINH — Hyperbolic Sine

SINH(*x*)
Function value: the hyperbolic sine for the specified argument.

SQRT — Square Root
SQRT(x) $x>0$
Function value: the positive square root of the specified positive argument.

STEP — Step
STEP($x1$, $x2$)
Function value:

$x1 \geq x2$	$x1 < x2$
STEP($x1$, $x2$) = 1	STEP($x1$,$x2$) = 0

SUM — Sum
SUM(*series*)
SUM(*listname*)
Function value: the sum of the values in the specified series or list or 0 if the specified listname contains no elements.

TAN — Tangent
TAN(x) x in radians
Function value: the tangent of the specified argument.

TANH — Hyperbolic Tangent
TANH(x)
Function value: the hyperbolic tangent for the specified argument.

Hardware Reference

This section describes how to use the TK!Solver program with your Radio Shack® TRS-80® Model 4 personal computer. The hardware information given here is intended only as a general guide. For specific information about your computer and printer, see your user manuals.

Requirements

To use the TK!Solver program, you need:

Hardware

- The TRS-80 Model 4 computer with at least 128K bytes of memory and one or more disk drives. A hard disk is optional. A hard disk must be formatted before using. Refer to the *TRS-80 Model 4 Hard Disk Start-Up Manual* for more information.
- Blank, formatted storage diskettes (optional). See the *TRS-80 Model 4 Disk System Owner's Manual* or Chapter 2 of the TK!Solver Instruction Manual for information on formatting diskettes.
- A Radio Shack parallel printer (optional). Refer to "Global Sheet Settings" later in this section for information on printer defaults in the TK!Solver program.

Software

- The TK!Solver program diskette. The program diskette contains all the TRSDOS® files needed to boot the system.

Files

Your TK!Solver program diskette contains program files and Instruction Files. You may make six backup copies of the TK!Solver program and one backup copy on a hard disk for use on a single computer.

Any other copying of the TK!Solver program diskette, the TK!Solver program files, or the TRSDOS System files is unauthorized and is a violation of copyright law and the TK!Solver license agreement.

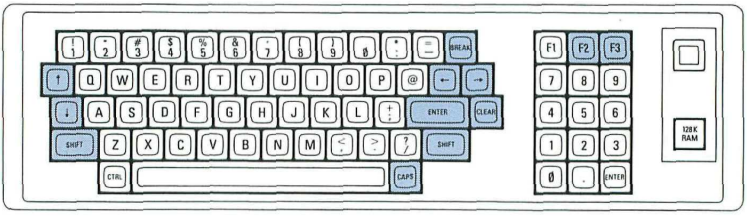
TK!Solver Keys

The Instruction Files are sample files that are used throughout the Instruction Manual. The names of these files are:

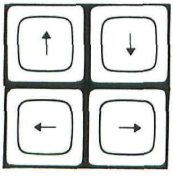
- COST/TK
- GRAVITY/TK
- TUNITS/TK
- LINEAR/TK
- MORTGAGE/TK
- NPOWER/TK
- FUNC/TK
- TFUNC/TK

TK!Solver Keys

The TK!Solver program uses certain keys to perform actions and commands. Listed below are descriptions of the TK!Solver keys and the names and symbols for the keys as used in this text.



Arrow keys



The keys marked with the up, down, left, and right arrows are called the Arrow keys. These keys are represented in the text by the symbols ⬆, ⬇, ⬅, and ➡. ⬅ and ➡ are on the right side of the alphabetic keys. ⬆ and ⬇ are located on the left side of the alphabetic keys. Pressing ⬅, ➡, ⬆, or ⬇ moves the cursor in the direction of the arrow.

Backspace key



The Backspace key is marked **F2** and is located at the top of the numeric keypad on the right side of the keyboard. It is represented in the text by the symbol ◄. Pressing ◄ erases the character to the left of the cue.

Break key



The Break key is located at the top right of the keyboard and is marked **BREAK**. Pressing this key breaks out of or cancels the current entry. It is represented in the text by the symbol ⊗.

Clear key



The Clear key is located on the right side of the keyboard and is marked **CLEAR**. Hold down the Clear key to type a command or to display special symbols on the screen.

In the TK!Solver program, use the Clear key to type the following special characters:

To type:	Hold down:	and Press:
^ (exponentiation symbol)	Clear	;
\ (backslash)	Clear	/
_ (underscore)	Clear	Enter

Edit key



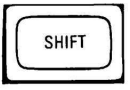
The Edit key calls the TK!Solver Editor. The Editor allows you to change an entry. The Edit key is labeled **F3** on your keyboard and is located above the numeric keypad on the right side of your keyboard. The F3 key is represented in this text by the symbol ⊠.

Enter key



The Enter key is marked **ENTER** and is a large key located on the right side of the alphabetic keys. It is represented in the text by the symbol ♦. Pressing ♦ enters information at the current cursor position or completes a command.

Shift keys



There are two Shift keys, one on either side of the bottom row of alphabetic keys. The Shift keys are marked **SHIFT**. Hold down a Shift key to type uppercase characters and the special symbols shown on the top half of the number keys.



To lock in uppercase letters, press the **CAPS** key to the right of the space bar. To return the keyboard to both uppercase and lowercase characters, press the CAPS key again.

Global Sheet Settings

The TK!Solver Global Sheet contains a list of default settings for your printer. The following two default printer settings are set by TRSDOS:

- Printer Device or Filename: *PR
- Line End: System-defined

The default for the **Printer Device or Filename** field is the TRSDOS device specification ***PR**. If you leave *PR in this field, you can send information contained in a TK!Solver model to a printer. See "Devices" later in this section for more information about device specifications.

If you want to alter the way a file is printed, use the TRSDOS command FORMS. See the *TRS-80 Model 4 Disk System Owner's Manual* for details.

The default for the **Line End** field is **System-defined (S)**. This means that TRSDOS already knows when to end a line. You cannot change the value of this field.

Printer Setup Strings

A setup string assigns printing options such as underlining and boldfacing. Enter a setup string into the **Printer Setup String** field, located on the TK!Solver Global Sheet.

Some examples of printer strings for the TRS-80 Model 4 and the Radio Shack DMP 200 printer are listed below.

- Underline ON: ^H0F
- Underline OFF: ^H0E
- Boldface ON: ^H1B^H1F
- Boldface OFF: ^H1B^H20
- Form Feed: ^H0C
- Correspondence Quality Printing: ^H1B^H12

- Standard Character Printing: ^H1B^H13
- Condensed Character Printing: ^H1B^H14
- Compressed Character Printing: ^H1B^H17

For further information on setup strings, see the section "Global Sheet" in Part II of the TK!Solver Reference Manual and the user's guide for your printer.

Filenames

Filenames have certain restrictions placed on them by TRSDOS. These restrictions are:

- Filenames must be one to eight characters long.
- Filenames may contain any uppercase or lowercase letter and any number from 0 through 9.
- Filenames must begin with a letter.
- Filenames cannot contain any other characters.

File Specifications

File specifications are added to a filename to identify or locate a file. In the TK!Solver program, you use file specifications when naming a file for the Storage command.

You do not always have to use these specifications when you name a file. If you leave out a specification, the TK!Solver program uses a default. These defaults are described below. The file specifications used by TRSDOS include the drive name, the file extension, the password, and the device. The general form for the TRSDOS file specifications is:

filename/extension.password:device

Each of these specifications is explained below.

Extension

An extension identifies different types of files. Extensions identify BASIC files, text files, TK!Solver files, and other application files. An extension is one to three alphanumeric characters preceded by a slash (/). The first character of an extension must be a letter.

The TK!Solver program uses three different types of extensions. You do not have to specify the extension when you load or save a file in the TK!Solver program.

Files saved using the Storage command with the S, V, U, and F options are automatically given the extension /TK. Files loaded with the L option must have filenames with the extension /TK, although you do not have to specify this extension.

Files saved using the DIF™ option (#) of the Storage command are automatically given the extension /DIF. Files loaded using the DIF option must have filenames with the extension /DIF.

Files saved using the Print Command (/P) are automatically given the extension /PRF.

Passwords

A password, when added to a filename, restricts other users from looking at or changing the file, unless they know the password. Passwords are one to eight alphanumeric characters preceded by a period (.). The first character of a password must be a letter. When you name a file, add the password after the filename or after its extension.

You do not have to include a password when creating a file in the TK!Solver program. However, if an existing file already has a password, you must include the password after the filename when you load or save the file.

Devices

TRSDOS has two different kinds of devices: physical and logical. A physical device is a piece of hardware, such as a disk drive, a keyboard, or a printer. The disk drives are identified by the numbers 0 through 9. You can have up to two disk drives installed in the computer case next to the screen. The bottom drive is called drive 0 and the drive above it is called drive 1. A hard disk is also considered a drive and is called drive 0 after it is installed.

A logical device is the connection between TRSDOS and the drives, a printer, or other physical devices. TRSDOS uses a device name, also called **devspec**, to identify a connection between phys-

ical devices. The devspec is a two-character abbreviation preceded by an asterisk (*). The TRSDOS devspecs are listed below:

*DO	Display Output (video)
*JL	Job Log
*KI	Keyboard Input
*PR	Printer
*SO	Standard Output
*SI	Standard Input

By separating the connection of a physical device from the device itself, you can sometimes substitute one physical device for another. You can also substitute a file for a physical device.

In the TK!Solver program, the devspec ***PR** is the default setting for the Printer Device or Filename field on the Global Sheet. If you replace *PR in this field with a filename, the information that would have been printed will be saved to that filename instead.

System Notes

- To load the TK!Solver program, type **TK** and press ♦.
- To quit the TK!Solver program, type **/Q**. Confirm this command by typing **Y**.
- To use the TK!Solver program, you must have a copy of the file TK/OVL available on a diskette in a drive or on a hard disk. If you want to use the Help Facility, you must also have a copy of the file TK/HLP.
- Any diskette you have in drive 0 must contain the following TRSDOS System files: SYS0/SYS through SYS4/SYS, SYS6/SYS, SYS7/SYS, and SYS10/SYS through SYS13/SYS.
- If you use the Save option of the Storage command to save a file in the TK!Solver program and do not specify a drive number, the TK!Solver program may cancel the Storage command and display the following message:

```
Disk Space or Disk Directory Full
```

This message refers to one of the following problems:

—there is not enough space on any of the diskettes

—all the diskettes in the drives are write-protected

Although you were not able to save the file to any of the diskettes currently in the drives, the TK!Solver program still contains the information.

Before saving the file again, check to see if the diskette that is to contain the file is write-protected. A write-protected diskette is one that has either no square notch on the left or a metallic tab covering the notch.

If the diskette has a write-protect tab, remove the tab and reinsert the diskette. If the diskette is notchless, replace the diskette with a notched formatted diskette. Then use the Save option of the Storage command again to save the file on the unprotected diskette.

If the diskette is not write-protected, then it does not have enough space left to contain the file to be saved. Replace this diskette with another notched, unprotected diskette and save the file.

- In the TK!Solver program, you can use uppercase or lowercase letters for commands and built-in functions. However, the program does distinguish between uppercase and lowercase letters in all other names, including variables, units, user functions, and lists.
- The error signal is a beep. It is sometimes accompanied by a message on the Prompt/Error line.
- The numeric range for your computer is any number with an absolute value between 1E-120 and 1E120.
- The TK!Solver program supports the TRSDOS wildcard $. When used with the Load or Save command, $ replaces one character in a filename or its specifications. Refer to the *TRS-80 Model 4 Disk System Owner's Manual* for more information about this wildcard.

TK!Solver Files

This appendix describes the types of files that can be created using the TK!Solver program. These files are all created from the program onto the operating system level. (See the Hardware Reference section for the filename conventions for your computer.)

Print Files

A print file is created with the Print command (/P) when a filename is specified in the Printer Device or Filename field on the Global Sheet. Each print command overwrites the previous contents of the file, so the filename should be changed on the Global Sheet if the results of more than one Print command are to be saved.

A print file contains data from a single TK!Solver sheet or subsheet and looks similar to a sheet or subsheet displayed on the screen, except that it can display as much as an entire sheet or as little as a single row. A print file for a subsheet begins with a heading that identifies the variable, list, or user function name on the first line of the file. The options for printing defined on the Global Sheet are used in the creation of the print file. The lines printed into the file are displayed simultaneously on the screen with the defined output format as the file is created.

Plot and Table Files

A plot file is created from the Plot Sheet, and a table file from the Table Sheet, with the Action command (!) given two conditions:

- The Screen or Printer field on the Plot or Table Sheet is set to the printer option (P).
- A filename is specified in the Printer Device or Filename field on the Global Sheet.

Each Plot or Table action overwrites the previous contents of a file if a new name is not specified.

A plot file contains a single-page plot and a table file contains a single- or multiple-page table. The plot or table looks similar to a plot or table displayed only on the screen except that the options for printing defined on the Global Sheet are used in the plot or table file. The plot or table printed to the file is displayed simultaneously on the screen with the defined output format as it is created.

Storage Files

A storage file is created with the Save Model, Variable Save, Unit Save, and Function Save options of the Storage command (/S); a TK!SolverPack™ file has the same format. Either may be loaded into the program with the Load option of the Storage command. If the user tries to save data using an existing filename, the program prompts to see whether the file should be overwritten.

A storage file can be recognized from the operating system level by "TK!1" on the first line of the file.

A storage file saved with the Save Model option (/SS) contains information used to recreate a particular TK!Solver session, including all data, and the current screen display and cursor position. The other kinds of storage files save data related to variables, units, or functions, but do not save a cursor position or current display.

DIF Format Files

The DIF™ file format is a data interchange format developed by Software Arts, Inc., designed to provide a standard convention for the exchange of data between programs. A DIF file can be created with the DIF Save option of the Storage command (/S#S) within the TK!Solver program, or by another program that uses the DIF format.

DIF files are loaded into the TK!Solver program with the DIF Load option of the Storage command. If created by another program, they must be organized in a format consistent with the List Sheet and List subsheets. In order to load properly, all labels and string values in a DIF file must conform to the name item and symbolic value conventions for the TK!Solver program. All labels and string values cannot begin with a digit and can contain only letters, digits, and the special characters allowed in name items:

@ # $ % _

The TK!Solver program inserts the apostrophe at the beginning of string values to make them into symbolic values. If any invalid character exists in a label or string value, the name or symbolic value loaded into a field is truncated to include only the valid characters.

A TK!Solver DIF Save file contains list information from the List Sheet and List subsheets only and must be saved from the List Sheet.

A DIF file can be recognized from the operating system level by the Header Item label "TABLE" on the first line of the file and others such as "VECTOR", "TUPLE", and "DATA" in the top portion.

The DIF Clearinghouse coordinates and distributes information about DIF and commercially available programs that use DIF. For information, write to:

DIF Clearinghouse
PO Box 638
Newton Lower Falls, MA 02162

The clearinghouse would like to add to its list of programs that support DIF. No royalties or fees are charged for using DIF, but the clearinghouse makes three requests of authors of commercially available programs using DIF:

- Please send a one-page description of your program to the DIF Clearinghouse. Describe what the program does, what computers it runs on, how it uses DIF, and how it can be purchased.
- Because DIF is a trademark of Software Arts Products Corp., the "tm" designation must be used if DIF is mentioned in any material you publish about your program. The material should also include the statement "DIF is a trademark of Software Arts Products Corp."
- If you are unable to test your program with the TK!Solver and VisiCalc® programs, please write the DIF Clearinghouse before advertising your product as supporting DIF. Any program employing DIF should be capable of transferring data to and from both programs.

Unit Name Conventions

The following abbreviations for the names of standard units of measurement are used in TK!SolverPack™ models and are recommended conventions for use with the TK!Solver program (although the program accepts any unit names the user chooses). The abbreviations are given here in uppercase and lowercase to show their use on computers that use both cases. No periods are used.

Unit Abbreviations

Length	
Inch	in
Foot	ft
Yard	yd
Mile	mi
Angstrom	Ang
Micron	um
Millimeter	mm
Centimeter	cm
Meter	m
Kilometer	km
Area	
Square meter	m^2
Hectare	ha
Acre	acre
Volume	
Ounce	fl_oz
Pint	pt
Quart	qt
Gallon	gal
Cubic centimeter	cm^3
Milliliter	ml
Liter	l

Weight, Mass	
Ounce	oz
Ounce, troy	oz__t
Pound	lb
Pound, troy	lb__t
Ton, long	ton__l
Ton, short	ton
Milligram	mg
Gram	g
Kilogram	kg
Metric ton	t
Slug	slug
Time	
Millisecond	ms
Second	s
Minute	min
Hour	h
Day	day
Week	wk
Month	mo
Year	yr
Temperature	
Kelvin	K
Degree Centigrade	oC
Degree Fahrenheit	oF
Degree Rankine	oR
Planar Angle	
Radian	rad
Degree	deg
Minutes of arc	'
Seconds of arc	"
Pressure	
Atmosphere	atm
Centimeters of mercury	cm__Hg
Feet of water	ft__H2O
Pounds per square inch	lb/in^2
Pascal	Pa

Unit Name Conventions

Energy, Work, Heat, Force, Power, Frequency	
British thermal unit	Btu
Calorie	cal
Kilocalorie	kcal
Erg	erg
Horsepower	hp
Watt	W
Kilowatt	kW
Joule	J
Dyne	dyn
Newton	N
Hertz	Hz
KiloHertz	kHz
MegaHertz	MHz

Electricity	
Ampere	A
Milliampere	mA
Volt	V
Ohm	Ohm

Miscellaneous	
Percent	%
United States dollars	$
Deutsche mark	DM

Certain conventions have been devised and are recommended to maintain consistency in the development of new abbreviations:

- All units raised to a power are expressed as exponents of the standard abbreviations. For example:
 A square inch is expressed as in^2
 A cubic yard is expressed as yd^3
- The product of two units is symbolized by an asterisk between the two unit names. For example:
 An inch-pound is expressed as in*lb
 A kilowatt-hour is expressed as kW*h
- All units derived from the division of two units are expressed with a slash between them. For example:
 Miles per hour is expressed as mi/h (NOT mph)
- Underscores are used for clarity in certain combined or qualified unit names. For example:
 A fluid ounce is expressed as fl_oz
- Compound unit names are formed by combining the above principles. For example:
 Pounds per square inch is expressed as lb/in^2 (NOT psi)

Prompt Messages

Startup Screen:

The following prompt message appears on the bottom of the initial startup screen of the TK!Solver program on most computers:

******* Press ENTER to Start *******

Cause: Typing the command to begin the TK!Solver program, with a disk specifier if necessary, from the operating system level of a personal computer.

Actions: Pressing the Enter key to begin the program (to stop the program after pressing the Enter key, wait until the TK!Solver program finishes loading and type **/QY**); pressing the Break key to return immediately to the operating system level.

Prompt/Error Line:

The following prompt messages appear on the Prompt/Error line. Any command can be cancelled and the prompt erased by pressing the Break key.

/ Command: B C D E I L M P Q R S W !

Cause: Typing /.

Actions: Choosing one of the Slash commands:

B	Blank command
C	Copy command
D	Delete command
E	Edit Field command
I	Insert command
L	List command
M	Move command
P	Print command
Q	Quit command
R	Reset command
S	Storage command
W	Window command
!	Solve command

Cancelling the command by pressing the Break key or typing any character that is not a valid Slash command.

Blank: Point to last field (*Pointer Indicator*)

Cause: Typing the Blank command (/B) with the cursor on the first item to blank.

Actions: Pointing to the last item to blank in the same column, if necessary, by using the Arrow keys and the Goto command (:) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.)

Copy: Point to destination (*Pointer Indicator*)

Cause: Typing the Copy command (/C) with the cursor on the item to be copied.

Actions: Pointing to the destination field by using the Arrow keys, the Goto command (:), and the Switch command (;) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.) If the copied item is invalid in the new location, an error message appears on the Prompt/Error line describing the type of error, and the Editor is called automatically to edit the item.

DIF Load: Filename:

Cause: Typing the DIF Load option of the Storage command (/S#L).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

DIF Save: Filename:

Cause: Typing the DIF Save option of the Storage command (/S#S), pointing to the last list in the block of lists to save, and completing the command with the Enter key.

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

DIF Save: Point to last list (*Pointer Indicator*)

Cause: Typing the DIF Save option of the Storage command (/S#S) with the cursor on the first list to save on the List Sheet.

Actions: Pointing to the last list to save by using the Arrow keys and Goto command (:) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.) A prompt appears for a filename when the pointing is complete.

DIF Storage: S L

Cause: Typing the DIF option of the Storage command (/S#).

Actions: Choosing one of the List command options:

S Save
L Load

Cancelling the command by typing any character that is not a valid option.

Delete: Filename:

Cause: Typing the Delete option of the Storage command (/SD).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Delete filename: *filename*

Cause: Typing the Delete option of the Storage command (/SD) and calling the Directory Scrolling feature by pressing the Right Arrow key or entering a partial file specification with the Right Arrow key.

Actions: Pressing the Right Arrow key to display the next filename until the appropriate filename appears and then pressing the Enter key to select it; cancelling the command by using the Right Arrow key past the last filename or pressing the Break key.

Delete filename: *filename* **: Y N**

Cause: Typing the Delete option of the Storage command (/SD) and providing a filename by entering it or selecting it with the Directory Scrolling feature.

Actions: Typing **Y** to delete the file; typing **N** to cancel the command.

Delete: Point to last row (*Pointer Indicator*)

Cause: Typing the Delete command (/D) with the cursor on the first row to delete.

Actions: Pointing to the last row to delete, if necessary, by using the Arrow keys and the Goto command (:) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.)

Fill List: Y N

Cause: Typing the Action command (!) with the cursor on a List subsheet or on the domain or range list of a User Function subsheet with a numeric value in the first and last elements to be filled in.

Actions: Typing **Y** to fill the list; typing **N** to cancel the command.

Function Save: Filename:

Cause: Typing the Function Save option of the Storage command (/SF).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Get from element:

Cause: Typing the Get option of the List command (/LG).

Actions: Typing an integer or integer expression to indicate the element number from which to retrieve values for all associated lists, and pressing the Enter key to complete the command.

Goto: Destination or search:

Cause: Typing the Goto command (:).

Actions: Using the Destination option by providing a row number and/or label letter to move the cursor to the specified destination, and pressing the Enter key to complete the command; using the Search option by typing a quotation mark (") followed by an exact match of the portion of an item displayed in a field in the same column, and pressing the Enter key to complete the command.

Help: ? or topic:

Cause: Typing the Help command (?) or completing a previous Help request.

Actions: Typing **?** to display Help Facility instructions; typing a topic name and the Enter key to search for a topic; pressing the Enter or Break key to exit the Help Facility.

List: ! B P G

Cause: Typing the List command (/L).

Actions: Choosing one of the List command options:

!	Solve
B	Block
P	Put
G	Get

Cancelling the command by typing any character that is not a valid option.

List Block: First element:

Cause: Typing the Block option of the List command (/LB).

Actions: Typing an integer or integer expression to indicate the beginning of the block of elements to be processed from all associated lists, and pressing the Enter key. A prompt appears for the final element number.

List Block: Last element:

Cause: Typing the Block option of the List command and providing a first list element number (/LB*element* ♦).

Actions: Typing an integer or integer expression to indicate the end of the block of elements to be processed from all associated lists, and pressing the Enter key to complete the command.

Load filename: *filename*

Cause: Typing a Load option of the Storage command (/SL or /S#L) and calling the Directory Scrolling feature by pressing the Right Arrow key or entering a partial file specification with the Right Arrow key.

Actions: Pressing the Right Arrow key to display the next filename until the appropriate filename appears and pressing the Enter key to select it; cancelling the command by using the Right Arrow key past the last filename or pressing the Break key.

Load: Filename:

Cause: Typing the Load option of the Storage command (/SL).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Move: Point to destination (*Pointer Indicator*)

Cause: Typing the Move command (/M) with the cursor on the row to be moved.

Actions: Pointing to the destination row by using the Arrow keys and Goto command (:) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.)

Next page: Y N

Cause: Typing a topic name in the Help Facility that takes more than one page to display.

Actions: Typing **Y** to display the next page; typing **N** to redisplay the Help topic prompt.

No topic *topic*; **More help: Y N**

Cause: Typing a topic that does not exist in the Help Facility.

Actions: Typing **Y** to redisplay the Help topic prompt and entering another topic; typing **N** to exit the Help Facility.

Overwrite filename: *filename* **: Y N**

Cause: Typing a Save option of the Storage command (/SS, /SV, /SU, /SF, or /S#S) and providing a filename that already exists by entering it or selecting it with the Directory Scrolling feature.

Actions: Typing **Y** to overwrite the file; typing **N** to cancel the command.

Print: Point to last line (*Pointer Indicator*)

Cause: Typing the Print command (/P) with the cursor on the first row to be printed.

Actions: Pointing to the last row to print by using the Arrow keys and Goto command (:) in any combination, and completing the command with the Enter key. (The Pointer Indicator changes to indicate the cursor position during pointing.)

Put to element:

Cause: Typing the Put option of the List command (/LP).

Actions: Typing an integer or integer expression to indicate the element number in which to store values from the Variable Sheet for all associated lists, and pressing the Enter key to complete the command.

Quit: Y N

Cause: Typing the Quit command (/Q).

Actions: Typing **Y** to exit the program and delete all data; typing **N** or pressing the Break key to cancel the command and remain in the program.

Reset all: Y N

Cause: Typing the All option of the Reset command (/RA).

Actions: Typing **Y** to clear the program; typing **N** to cancel the command.

Reset current sheet: Y N

Cause: Typing the Sheet option of the Reset command (/RS).

Actions: Typing **Y** to clear the current sheet; typing **N** to cancel the command.

Reset variable values: Y N

Cause: Typing the Variable option of the Reset command (/RV).

Actions: Typing **Y** to reset all variable values; typing **N** to cancel the command.

Reset: V S A

Cause: Typing the Reset command (/R).

Actions: Choosing one of the Reset command options:

V	Variable
S	Sheet
A	All

Cancelling the command by typing any character that is not a valid option.

Save: Filename:

Cause: Typing the Save Model option of the Storage command (/SS).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Save filename: *filename*

Cause: Typing a Save option of the Storage command (/SS, /SV, /SU, /SF, or /S#S) and calling the Directory Scrolling feature by pressing the Right Arrow key or entering a partial file specification with a Right Arrow key.

Actions: Pressing the Right Arrow key to display the next filename until the appropriate filename appears and pressing the Enter key to select it; cancelling the command by using the Right Arrow key past the last filename or pressing the Break key.

Sheet: V R U G L F P T

Cause: Typing the Select command (=).

Actions: Choosing one of the Select command options:

V	Variable Sheet
R	Rule Sheet
U	Unit Sheet
G	Global Sheet
L	List Sheet
F	User Function Sheet
P	Plot Sheet
T	Table Sheet

Cancelling the command by typing any character that is not a valid option.

Storage: L S V U F # D

Cause: Typing the Storage command (/S).

Actions: Choosing one of the Storage command options:

L	Load File
S	Save Model
V	Variable Save
U	Unit Save
F	User Function Save
#	DIF Storage
D	Delete File

Cancelling the command by typing any character that is not a valid option.

Unit Save: Filename:

Cause: Typing the Unit Save option of the Storage command (/SU).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Variable Save: Filename:

Cause: Typing the Variable Save option of the Storage command (/SV).

Actions: Typing a file specification followed by the Enter key, or calling the Directory Scrolling feature with the Right Arrow key to select a filename.

Window: 1 V R U G L F P T

Cause: Typing the Window command (/W).

Actions: Choosing one of the Window command options:

1	One Window
V	Variable Sheet
R	Rule Sheet
U	Unit Sheet
G	Global Sheet
L	List Sheet
F	User Function Sheet
P	Plot Sheet
T	Table Sheet

Cancelling the command by typing any character that is not a valid option.

Status Messages

Cleared Screen:
The following status message appears at the top of a blank screen:

```
Loading the TK!Solver program
 Copyright 1983 Software Arts, Inc.
```

Cause: Typing the Enter key from the TK!Solver startup screen and beginning the program.

Actions: Waiting until the message disappears from the screen before beginning to use the program.

Status Line:
The following status messages appear on the Status Line:

!

Cause: Entering information into any field that might require re-solution of the model.

Actions: Calling one of the solvers with the Action command (!) from the Variable Sheet or the Rule Sheet, or the Solve command (/!) from any sheet.

LOW

Cause: Any action that depletes remaining memory below 1K bytes.

Actions: Saving information with the Storage command (/S) and clearing some or all of a model with the Reset command (/R), or deleting list values with the Delete command (/D); performing actions which require minimal memory.

OUT

Cause: Any action that depletes remaining memory.

Actions: Saving information with the Storage command (/S) and clearing some or all of a model with the Reset command (/R), or deleting list values with the Delete command (/D); exiting the program with the Quit command (/Q).

Status Messages

Prompt/Error Line:
The following status messages appear on the Prompt/Error Line. Pressing the Break key cancels all actions unless noted otherwise.

Blanking fields

Cause: Typing the Blank command (/B), pointing to a block of rows, and pressing the Enter key to complete the command.

Deleting *filename*

Cause: Typing the Delete option of the Storage command (/SD), providing a filename by entering it or selecting it with Directory Scrolling, and confirming the deletion by typing **Y**.

Deleting rows

Cause: Typing the Delete command (/D), pointing to a block of rows, and pressing the Enter key to complete the command. The Break key does not interrupt this action.

Direct Solver

Cause: Typing the Action command (!) from the Variable Sheet or the Rule Sheet, or the Solve command (/!) from any sheet.

(Edit)

Cause: Typing the Edit Field command (/E) or the Edit command (⊠) to call the Editor. (This message also appears at the beginning of error messages that call the Editor automatically. See the appendix on Error Messages for details.)

Filling in list

Cause: Typing the Action command (!) from a List subsheet or User Function subsheet and typing **Y** to confirm.

For Help, type ?

Cause: Pressing the Enter key from the TK!Solver startup screen and waiting until the program loads. (This message is only a reminder on how to call the Help Facility and requires no action.)

Iterative Solver: *step* **out of** *estimate*

Cause: Typing the Action command (!) from the Variable Sheet or the Rule Sheet, or the Solve command (/!) from any sheet if a guess value is set for any variable on the Variable Sheet, or automatic iteration is turned on and first guess values are provided on Variable subsheets. The step of the iteration is displayed out of the number of iterations the program expects the solution to take. The estimate is revised during solution.

List Solver: *element*, **Direct Solver**

Cause: Typing the Solve option (/L!) or the Block option (/LB) of the List command to apply the Direct Solver for all associated lists in a model. A count is displayed for the list element currently being processed.

List Solver: *element*, **Iterative Solver:** *step* **out of** *estimate*

Cause: Typing the Solve option (/L!) or the Block option (/LB) of the List command to apply the Iterative Solver to all associated lists in a model if a guess value is set for any variable on the Variable Sheet, or automatic iteration is turned on and first guess values are provided on Variable subsheets. A count is displayed for the element currently being processed. The step of the iteration is displayed out of the number of iterations the program expects the solution to take. The estimate is revised during solution.

List Solver: *element*, **Iterative Solver:** *step*

Cause: The same message as above if too long to display on a screen width less than 80 characters.

Loading from *filename*

Cause: Typing a Load option of the Storage command (/SL or /S#L), and providing a filename by entering it or selecting it with Directory Scrolling.

Resetting all variables

Cause: Typing the Variable option of the Reset command (/RV) and confirming it by typing **Y**. The Break key does not interrupt this action.

Resetting current sheet

Cause: Typing the Sheet option of the Reset command (/RS) and confirming it by typing **Y**. The Break key does not interrupt this action.

Resetting everything

Cause: Typing the All option of the Reset command (/RA) and confirming it by typing **Y**. The Break key does not interrupt this action.

Saving to *filename*

Cause: Typing a Save option of the Storage command (/SS, /SV, /SU, /SF, or /S#S), pointing to the last list in the case of the DIF Save option, providing a filename by entering it or selecting it with Directory Scrolling, and confirming for overwriting if necessary by typing **Y**.

Searching the Help file

Cause: Typing a topic followed by the Enter key in the Help Facility.

Rule Sheet: Status Column:
The explanation of a status message in a Status field on the Rule Sheet may be viewed on the Status line at the top of the screen by moving the cursor to the Status field in question. (See the appendix on Error Messages for the error messages that appear in Status fields.)

The following status messages are available:

Comment

Cause: The Rule field on the same row contains only a comment, beginning with a quotation mark ("). No character displays in the Status field.

Satisfied

Cause: The Rule field on the same row contains an equation evaluated by either of the solvers during the previous solution and determined to have a supplied or derived value for each variable. No character displays in the Status field.

*** Unsatisfied**

Cause: The Rule field on the same row contains an equation that was not evaluated by either of the solvers during the previous solution. An asterisk displays in the Status field.

Error Messages

Error messages are organized according to where they appear on the TK!Solver screen: on the startup screen, on the Prompt/Error line, in the Status fields on the Variable and Rule sheets, at the bottom of the screen, in value locations, and on a cleared screen. The explanations below are approximate and intended to serve as a guide only.

Error messages are documented generally for all machines that use diskettes. An error message may be displayed on your computer that is not listed below. See the Hardware Reference section for further information on machine-specific messages.

If a message is displayed that does not appear anywhere in the Reference Manual, or a condition other than the one described produced the error, write down any pertinent information and contact your dealer.

Startup Screen:
The following error messages appear at the beginning of the program on the initial screen on most implementations:

******* Disk Read Error *******
... Press ENTER to Exit ...

Cause: Attempting to use a worn or damaged TK!Solver program diskette, or encountering a hardware error.

Actions: Pressing the Enter key and using your backup TK!Solver diskette or obtaining a new TK!Solver program diskette. (See your license agreement for details.) Consulting your hardware operations manual for hardware testing procedures; noting the message above and contacting your dealer.

******* Invalid Program Disk *******
... Press ENTER to Exit ...

Cause: Attempting to use a TK!Solver program diskette that has been overwritten or a diskette that contains the results of an attempt to copy the entire TK!Solver program diskette.

Actions: Pressing the Enter key and obtaining a proper TK!Solver program diskette; reformatting the diskette before using it for other purposes. (See your license agreement for details.)

******* Insufficient Memory *******
... Press ENTER to Exit ...

Cause: Attempting to use a TK!Solver program diskette with insufficient memory available.

Actions: Contacting your dealer and obtaining more memory for your computer system. The TK!Solver program requires a recommended minimum memory configuration. (See the Hardware Reference section for details.)

Prompt/Error Line:

The following error messages appear on the Prompt/Error line. Expressions or entries that contain multiple errors are diagnosed for the first recognized error. Error messages that begin with "(Edit)" automatically call the Editor for error correction: use the Enter key to enter the corrections and exit the Editor; use the Break key to exit the Editor and return the previous correct item.

Deleted entity

Cause: Attempting to perform an action on a Variable, List, or User Function subsheet that is still displayed on the screen for a variable, list, or user function that has been deleted using **/D** or **/RSY**.

Actions: Returning to the corresponding sheet, entering a variable, list or user function name, and diving to a new subsheet.

Disk Data or I/O Error

Cause: Attempting an action that causes a hardware read/write error or attempting to use a worn or damaged diskette. (This condition should not occur.)

Actions: Trying another diskette; referring to your hardware operations manual for hardware diagnostic tests; noting this message and contacting your computer dealer.

Disk Hardware Error

Cause: Attempting an action that causes a hardware read/write error. (This condition should not occur.)

Actions: Trying another diskette; referring to your hardware operations manual for hardware diagnostic tests; noting this message and contacting your computer dealer.

Disk is Protected

Cause: Attempting a print or storage action to a protected diskette.

Actions: Specifying another filename or drive specifier, or inserting another diskette.

Disk Related Software Error

Cause: Attempting an action that causes an operating system or program error.

Actions: Checking that there is no write-protect tab on the diskette to be written to, specifying another disk drive or filename, or trying another storage diskette; saving a copy of the model and a list of the actions that produced the error, noting the error message above, and contacting your dealer.

Disk Space or Disk Directory Full

Cause: Attempting a print or storage action to a diskette that has reached the limit for the number of files or amount of data permitted.

Actions: Inserting another diskette or specifying a different disk drive; deleting filenames with the Delete option of the Storage command (/SD) or at the operating system level. (A partial file may have been created on the original diskette and should be deleted with the Delete option (/SD) of the Storage command or on the operating system level.)

(Edit) Built-in function already exists

Cause: Attempting to enter a built-in function name in the Name field on the User Function Sheet. Built-in functions are case insensitive, so both uppercase and lowercase names are restricted.

Actions: Entering a non-reserved function name. (See the Built-in Functions section for the names of the built-in functions available in the TK!Solver program that cannot be used.)

(Edit) Equal sign within parentheses

Cause: Attempting to enter a rule that contains an equal sign within a set of parentheses or a left parenthesis not closed before the equal sign.

Actions: Editing the rule so that all sets of parentheses are closed on either side of the equal sign.

(Edit) Expression too complex

Cause: Attempting to enter a rule that contains too many repeated positive or negative signs or more than ten levels of parentheses.

Actions: Rewriting the rule more simply or dividing a rule into less complex rules, with extra variables if necessary.

(Edit) Function already exists

Cause: Attempting to enter a user function name in the Name field on the User Function Sheet that is the same as a previously entered user function name.

Actions: Entering a unique user function name.

(Edit) Invalid column width

Cause: Attempting to enter an integer value that is invalid for the Width field on the Table Sheet.

Actions: Entering an integer entry or expression between 3 and 128 (or the number of character positions to be printed in each column, less than the printed page or screen width).

(Edit) Invalid comparison tolerance

Cause: Attempting to enter a numeric value in the Comparison Tolerance field on the Global Sheet that multiplied by the typical value is equal to or greater than 1E120.

Actions: Entering a numeric value or expression that when multiplied by the typical value is less than 1E120. (See the Overview section for more information on valid numeric value entries.)

(Edit) Invalid first element

Cause: Attempting to enter an integer value that is invalid for the First field on the Table Sheet.

Actions: Entering an integer entry or expression between 1 and the integer limit for your machine (or any number up to the number of elements in the longest list).

(Edit) Invalid form length

Cause: Attempting to enter an integer value that is invalid for the Form Length field on the Global Sheet.

Actions: Entering an integer entry or expression between 6 and 1000 (or the number of lines per page on a printer page).

(Edit) Invalid function name

Cause: Attempting to enter a rule that contains a number before a left parenthesis where a function name is expected.

Actions: Editing the rule so that it contains an operator or a built-in or defined user function name before the parenthesis.

(Edit) Invalid integer

Cause: Attempting to enter a value or expression that does not evaluate to an integer in an integer entry field, or is outside machine limits.

Actions: Entering a valid integer entry or expression. (See the Overview and Hardware Reference sections for more information on valid integer entries.)

(Edit) Invalid iteration count

Cause: Attempting to enter an integer value that is invalid for the Iteration Count field on the Global Sheet.

Actions: Entering an integer entry or expression between 1 and 10000 (or the maximum number of iterations before the Iterative Solver stops).

(Edit) Invalid left margin

Cause: Attempting to enter an integer value that is invalid for the Left Margin field on the Global Sheet.

Actions: Entering an integer entry or expression that is between 0 and 200 (or the number of character positions to indent before printing the first character).

(Edit) Invalid name

Cause: Attempting to enter an invalid name in a Name field on the Variable Sheet, List Sheet, or User Function Sheet, or any place a list name can be assigned and created.

Actions: Entering a valid name entry. (See the Overview section for more information on valid name entries.)

(Edit) Invalid numeric value

Cause: Attempting to enter a symbolic value in an integer or numeric value field.

Actions: Entering a valid integer or numeric entry or expression. (See the Overview section for more information about valid integer or numeric entries.)

(Edit) Invalid operator

Cause: Attempting to enter a rule that contains an invalid character where an operator is expected. Valid characters are numbers, letters, the operators, an equal sign, sets of parentheses, apostrophes, commas within function argument lists, and special characters allowed in name entries:
@ # $ % _

Actions: Editing the rule so that it contains only valid characters. (See the Overview and Mathematics sections for more information on valid rules.)

(Edit) Invalid page length

Cause: Attempting to enter an integer value that is invalid for the Printed Page Length field on the Global Sheet.

Actions: Entering an integer entry or expression that is between 6 and 1000 (or the number of lines available for printing on each output page).

(Edit) Invalid page width

Cause: Attempting to enter an integer value that is invalid for the Printed Page Width field on the Global Sheet.

Actions: Entering an integer entry or expression that is between 6 and 200 (or the number of character positions available to the program for printing across an output page).

(Edit) Invalid symbolic value

Cause: Attempting to enter a rule that contains a number preceded by an apostrophe.

Actions: Editing the rule so that the apostrophe is omitted or the symbolic value begins with an apostrophe and a letter or @ # $ % _ (see the Mathematics section for more information about valid symbolic values).

(Edit) Invalid typical value

Cause: Attempting to enter a numeric value in the Typical Value field on the Global Sheet that multiplied by the comparison tolerance is equal to or greater than 1E120.

Actions: Entering a numeric value or expression in the Typical Value field on the Global Sheet that when multiplied by the comparison tolerance is less than 1E120. (See the Overview section for more information on valid numeric value entries.)

(Edit) List already exists

Cause: Attempting to enter a list name in a Name field on the List Sheet that is the same as a previously entered list name.

Actions: Entering a unique list name.

(Edit) Misplaced comma

Cause: Attempting to enter a rule that contains a comma between values or names that are not function arguments.

Actions: Editing the rule so that the only use of commas is to separate arguments of a function.

(Edit) Missing equal sign

Cause: Attempting to enter a rule that does not contain an equal sign.

Actions: Editing the rule so that it contains an equal sign.

(Edit) Missing function arguments

Cause: Attempting to enter a rule that contains a function with too few arguments.

Actions: Editing the rule so that it contains functions with the proper number of arguments: user functions take one argument; built-in functions take predetermined numbers of arguments. (See the Mathematics and Built-in Functions sections for more information on valid function argument formats.)

(Edit) Missing name

Cause: Attempting to enter a rule that contains an apostrophe followed by a character other than a letter or @ # $ % _

Actions: Editing the rule so that an apostrophe is followed by a valid name beginning with a letter or valid character or removing the apostrophe. (See the Overview section for more information on valid name entries.)

(Edit) Missing operand

Cause: Attempting to enter a rule that contains a dangling operator or equal sign; or a constant value that is outside the numeric limits for the program.

Actions: Editing the rule so that plus and minus signs precede names, values, or parenthetical expressions; and each equal sign or other operator has a name, value, or valid expression on either side of it; assigning a constant value with an absolute value less than 1E120.

(Edit) Missing operator

Cause: Attempting to enter a rule that contains two parenthetical expressions unrelated by an operator.

Actions: Editing the rule so that it contains an operator between the expressions within the two sets of parentheses.

(Edit) Too many equal signs

Cause: Attempting to enter a rule that contains more than one equal sign.

Actions: Editing the rule so that it contains only one equal sign or splitting the rule into two or more rules, each with one equal sign.

(Edit) Too many function arguments

Cause: Attempting to enter a rule that contains a function with too many arguments.

Actions: Editing the rule so that it contains functions with the proper number of arguments: user functions take one argument; built-in functions take predetermined numbers of arguments. (See the Mathematics and Built-in Functions sections for more information on valid function argument formats.)

(Edit) Undefined function

Cause: Attempting to enter a rule that contains a variable name or an undefined user function name before a left parenthesis.

Actions: Editing the rule so that it contains an operator before the parenthesis or defining a user function before entering it in a rule; checking the spelling of the intended built-in or user function name.

(Edit) Unevaluated expression

Cause: Attempting to enter an expression that contains a syntax error or a value outside the machine number limits in any value field.

Actions: Entering a value with an absolute value less than 1E120; entering a corrected expression; checking the expression by making it a rule on the Rule Sheet and looking at the Status field message for a more explicit error message.

(Edit) Unmatched left parenthesis

Cause: Attempting to enter a rule that contains an unmatched opening parenthesis.

Actions: Editing the rule so that it contains an equal number of right and left parentheses.

(Edit) Unmatched right parenthesis

Cause: Attempting to enter a rule that contains an unmatched closing parenthesis.

Actions: Editing the rule so that it contains an equal number of right and left parentheses.

(Edit) Variable already exists

Cause: Attempting to enter a variable name in the Name field on the Variable Sheet that is the same as a previously entered variable name.

Actions: Entering a unique variable name.

End of File

Cause: Attempting to load an incomplete or improperly formed file with the Load file (/SL) or DIF Load (/S#L) option of the Storage command. (This condition should not occur.)

Actions: Creating a new copy of the file and repeating the load action.

File Does Not Exist

Cause: Attempting to load a file that does not exist with the Load file (/SL) or DIF Load (/S#L) options of the Storage command.

Actions: Beginning the command again and entering a properly spelled filename; using the Directory Scrolling feature of the Storage command to locate the files that exist on the current diskette or inserting the diskette that contains the specified filename. (See the Storage command in the Command section for more information on Directory Scrolling.)

File is protected

Cause: Attempting a print or storage action to a filename that is protected by the operating system.

Actions: Specifying another filename or drive specifier, or inserting another diskette.

Header too long

Cause: Attempting to produce a table with the Action command (!) on the Table Sheet, with the Vertical or Horizontal field set to the Horizontal option, when the header specified in the Header field or the default list name header is too long to display any values on the same line. The total number of characters in the list name or Header field plus the value of the Width field must be less than the screen width or the value in the Printed Page Width field on the Global Sheet.

Actions: Entering a shorter header; setting the Vertical option; decreasing the value in the Width field; or increasing the integer in the Printed Page Width field on the Global Sheet (if the Screen or Printer field is set to the Printer option).

Incomplete load

Cause: Attempting to load a damaged file with the Load options (/SL and /S#L) of the Storage command; or pressing the Break key during loading.

Actions: Beginning the Storage command again and allowing the file to load completely; recreating and resaving the data in the damaged file. If the file is damaged, the load is completed up to the damaged section.

Incomplete save

Cause: Attempting a save action that causes a disk error; pressing the Break key during the use of the Save options of the Storage command (/S).

Actions: Beginning the Storage command again; specifying a different disk drive or filename; checking if the diskette is full or has a write-protect tab; or inserting a different diskette. A partial file may be created on the original diskette and should be deleted with the Delete option of the Storage command (/SD) or on the system level.

Invalid DIF file

Cause: Attempting to load a non-DIF file with the DIF Load option of the Storage command (/S#L); attempting to load a file that does not conform to the DIF™ file format.

Actions: Beginning the Storage command again and using the Load option of the Storage command (/SL) instead of the DIF Load option (/S#L) if the file is a regular TK!Solver save file; resaving the file with the DIF Save option (/S#S) within the TK!Solver program, or with the program which produced the file, if the file is damaged; or correcting the program that produced the file to conform with the DIF format. (See the Appendix on the files used with the TK!Solver program for further information concerning DIF files.)

Invalid element

Cause: Specifying no element number; attempting to enter a value or expression that evaluates to a negative integer or zero, in response to the prompt of the Put (/LP) or Get (/LG) options of the List command.

Actions: Beginning the List command again and entering a valid positive integer value or expression. (See the Overview section for more information on valid integer entries.)

Invalid File Specification

Cause: Attempting to specify an invalid filename for one of the Storage command (/S) options.

Actions: Supplying a file specification valid for your operating system or computer. (See the Hardware Reference section or your computer manual for further information about valid file specifications.)

Invalid first element

Cause: Attempting to enter a value or expression that evaluates to a negative integer or zero in response to the prompt for the first element number by the Block option of the List command (/LB).

Actions: Beginning the List command again and entering a valid positive integer value or expression. (See the Overview section for more information on valid integer entries.)

Invalid integer

Cause: Attempting to enter a value or expression that does not evaluate to an integer, or an integer outside the valid integer limits on the computer being used, on the Prompt/Error line in response to the Block (/LB), Put (/LP), and Get (/LG) options of the List command.

Actions: Beginning the List command again and entering a valid integer entry or expression. (See the Overview and Hardware Reference sections for more information on valid integer entries.)

Invalid last element

Cause: Designating no input list; attempting to enter a value or expression that evaluates to an integer smaller than the value specified for the first element number in response to the prompt for the last element number by the Block option of the List command (/LB).

Actions: Beginning the List command again and entering a valid positive integer value or expression greater than the value specified for the first element number. (See the Overview section for more information on valid integer entries.)

Invalid storage file

Cause: Attempting to load a file with the Load option of the Storage command (/SL) that does not conform to the TK!Solver storage file format.

Actions: Beginning the Storage command again and selecting a valid storage file name; or recreating the data and saving it properly with one of the Save options of the Storage command (/S).

Line Too Long

Cause: Attempting to load a file that contains a line that is too long (this condition could be caused by removing a line end character with an editor on the operating system level); attempting to load an improperly formatted file.

Actions: Pressing the Break key to cancel the load action; trying to recreate the file and reload it using the same machine.

List Solver: No associated lists

Cause: Attempting to use the Solve option of the List command (/L!) when there are no lists associated with variables on the Variable Sheet.

Actions: Creating an associated list by typing the List option (L) in the Status field for at least one variable on the Variable Sheet and beginning the List command again.

List Solver: No input list values

Cause: Attempting to use the Solve option of the List command (/L!) when there is no list associated with an input variable or no associated list containing values.

Actions: Assigning at least one variable associated with a list as an input variable with the Input option (I) on the Status field on the Variable Sheet, or entering values on the associated List subsheet, and beginning the List command again; using the Block option of the List command (/LB) to generate output values.

Missing domain list

Cause: Attempting to enter a value in a domain element field on a User Function subsheet before a domain list name has been entered.

Actions: Entering a name in the Domain List field of the User Function subsheet before entering values.

Missing range list

Cause: Attempting to enter a value in a range element field on a User Function subsheet before a range list name has been entered.

Actions: Entering a name in the Range List field of the User Function subsheet before entering values.

No ! action for this sheet

Cause: Attempting to use the Action command (!) on the Unit, Global, List, or User Function Sheet, or a Variable subsheet.

Actions: Using the Solve command (/!) to solve the current model from the current sheet or the Action command (!) by moving the cursor to the appropriate sheet or subsheet: the Variable or Rule sheets to solve the current model; the List or User Function subsheets to fill in list values; the Table Sheet to produce a table; or the Plot Sheet to produce a plot.

No associated lists

Cause: Attempting to use the Put (/LP) or Get (/LG) option of the List command when there are no lists associated with variables on the Variable Sheet.

Actions: Creating an associated list by typing the List option (L) in the Status field for at least one variable on the Variable Sheet and beginning the List command again.

No matching filenames

Cause: Attempting to use a complete file specification or a partial file specification containing a computer system wildcard in response to one of the options of the Storage command (/S) that does not match any filename on the default or specified diskette.

Actions: Using a wildcard correctly to indicate known filenames; including the appropriate drive specifier; using the Directory Scrolling feature of the Storage command without a filename to locate existing files. (See the Storage command in the Command section for more information on Directory Scrolling.)

On file systems with file extensions, it may be necessary to specify an extension or provide a wildcard for the extension if it is different from the default file extension.

No points to plot

Cause: Attempting to produce a plot with the Action command (!) on the Plot Sheet with no X or Y list specified, or no values in the X or Y axis lists.

Actions: Entering an X axis list name and at least one Y axis list name; entering at least one value in the X axis list and at least one value in at least one Y axis list before plotting a graph.

No Such Device or Device Not Ready

Cause: Attempting a print or storage action to a drive specifier that does not exist or a device name that is not connected.

Actions: Specifying a valid drive specifier or device name for a connected device and repeating the action. (See your operations manual for further information about device names and drive specifiers.)

No topic ⟨topic⟩; More help: Y N

Cause: Attempting to locate a topic in the Help Facility that does not exist.

Actions: Typing **Y** to redisplay the Help topic prompt and entering another topic (such as "Topics" to provide a list of valid topic names); typing **N** or pressing the Break key to exit the Help Facility and check the TK!Solver manuals for detailed information about the program.

No values to print

Cause: Attempting to produce a table with the Action command (!) on the Table Sheet with no lists specified or no values in the specified lists.

Actions: Entering at least one list name in the List field and at least one value in at least one list and repeating the Action command.

No variable name

Cause: Attempting to enter a List (L) or Guess (G) option in the Status field on the Variable Sheet for an unnamed variable.

Actions: Entering a variable name in the Name field on the Variable Sheet before associating a list or setting a guess value.

Not on List Sheet

Cause: Attempting to use the DIF Save option of the Storage command (/S#S) without the cursor on a list row on the List Sheet.

Actions: Moving the cursor to any field on the row for the first list to be saved on the List Sheet and repeating the command.

Options: I O L G B

Cause: Attempting to type an invalid option letter on the Status field on the Variable Sheet.

Actions: Entering one of the valid options:

I	Input variable assignment
O	Output variable assignment
L	List association
G	Guess variable assignment
B	Blank value

Options: L C

Cause: Attempting to type an invalid option letter on the Line End (CR/LF or CR) field on the Global Sheet.

Actions: Entering one of the valid options:

L	Carriage Return/Line Feed line ending on output
C	Carriage Return line ending on output

Options: S P

Cause: Attempting to type an invalid option letter on the Screen or Printer field on the Plot or Table Sheet.

Actions: Entering one of the valid options:

S	Screen display
P	Printer or file output

Options: T S L

Cause: Attempting to type an invalid option letter on the Mapping field on the User Function Sheet or a User Function subsheet.

Actions: Entering one of the valid options:

T	Table Lookup
S	Step Interpolation
L	Linear Interpolation

Options: V H

Cause: Attempting to type an invalid option letter on the Vertical or Horizontal field on the Table Sheet.

Actions: Entering one of the valid options:

V	Vertically displayed table
H	Horizontally displayed table

Options: Y N

Cause: Attempting to type an invalid option letter on the Display Scale ON field on the Plot Sheet; or the Variable Insert ON, Intermediate Redisplay ON, Automatic Iteration ON, Page Breaks ON, and Page Numbers ON fields on the Global Sheet.

Actions: Entering one of the valid options:

Y	Yes
N	No

Out of memory

Cause: Attempting to perform an action that requires more than the remaining available memory or produces a program error.

Actions: Saving information with the Storage command (/S) and clearing some or all of a model with the Reset command (/R), or deleting list values with the Delete command (/D); exiting the program with the Quit command (/Q); noting the actions that produced the error and contacting your dealer.

Output field only

Cause: Attempting to type, edit, copy into, or blank fields used only by the program for output: Output fields on the Variable Sheet, Status fields on the Rule Sheet, Elements fields on the List Sheet, or the blank lines on the Global Sheet.

Actions: Moving the cursor to a field that accepts user input; blanking an Output field on the Variable Sheet with the Blank option (B) in the Status field on the same row, or moving the output value to the corresponding Input field with the Input (I) or Guess (G) option.

System error: parser error

Cause: Attempting an action that causes an unknown program error. (This condition should not occur.)

Actions: Saving a copy of the model and a list of the actions that produced the error, noting the error message above, and contacting your dealer.

System error: unknown message

Cause: Attempting an action that causes an unknown program error. (This condition should not occur.)

Actions: Saving a copy of the model and a list of the actions that produced the error, noting the error message above, and contacting your dealer.

Unable to open printer

Cause: Attempting a print action to a device name or filename that cannot be opened.

Actions: Specifying a different device name or filename in the Printer Device or Filename field on the Global Sheet.

Unblankable block

Cause: Attempting to blank a block of rows in more than one column, or lines and rows in both the Header and Field areas of a sheet with the same Blank command (/B).

Actions: Using the Blank command (/B) for one column or one area at a time.

Unblankable field

Cause: Attempting to blank a required entry field by copying a blank field from another location with the Copy command (/C) or by backspacing over all characters with the Edit (⊠) or Edit Field (/E) commands.

Actions: Replacing the current value or option with a different value or option.

Unblankable field in block

Cause: Blanking a block of fields that contains one or more required entry or output only fields; attempting to use the Blank command on a Variable, List, or User Function subsheet still displayed on the screen for a variable, list, or user function that has been deleted.

Actions: This message is a warning only: blankable fields are blanked, and all others remain unchanged.

Undefined list block

Cause: Attempting to perform the list fill-in action of the Action command (!) on a List or User Function subsheet when the first or last value is not specified or contains a symbolic value.

Actions: Entering a numeric value or expression for the first element in the list and defining the end of the list to be filled in with another numeric value or expression, and repeating the Action command.

Undeletable block

Cause: Attempting to use the Delete command (/D) to delete a block of lines and rows from the Field Area into the Header Area of a sheet.

Actions: Using the Delete command to delete rows in a Field Area only.

Unevaluated expression

Cause: Attempting to enter an expression that contains a syntax error or a value outside the machine number limits on the Prompt/Error line in response to a prompt for an integer value by the List command (/L).

Actions: Beginning the command again and entering a valid integer value or expression with an absolute value less than 1E120. (See the Overview and Hardware Reference sections for more information about valid integer entries.)

Unknown Error

Cause: Attempting an action that causes an unknown program error. (This condition should not occur.)

Actions: Saving a copy of the model and a list of the actions that produced the error, noting the error message above, and contacting your dealer.

UNKNOWN ERROR CLASS

Cause: Attempting an action that causes an unknown program error. (This condition should not occur.)

Actions: Saving a copy of the model and a list of the actions that produced the error, noting the error message above, and contacting your dealer.

Variable Sheet: Status Column:
An error symbol (>) displays in the Status field on the Variable Sheet for each variable identified by the solvers as being involved in the termination of a solution. The explanation of an error symbol may be viewed on the Status line at the top of the screen by moving the cursor to the Status field in question.

The following explanatory error messages are provided:

› Error

Cause: The variable on the same row is probably involved in the error conditions that terminated the solution.

Actions: Moving the cursor to the error symbols in the Status column on the Rule Sheet for more explicit error diagnostics.

› Overdefined

Cause: The variable on the same row is probably inconsistent with at least one other variable value in at least one equation.

Actions: Blanking one or more values on the Variable Sheet for the overdefined variable, or editing any rules marked "› Inconsistent" in the Status field on the Rule Sheet that assign the variable with contradictory values.

Rule Sheet: Status Column:
An error symbol (›) displays in the Status field on the Rule Sheet for each rule identified by the solvers as being involved in the termination of a solution. The explanation of an error symbol may be viewed on the Status line at the top of the screen by moving the cursor to the Status field in question. (See the appendix on Status Messages for the status messages that appear in Rule Status fields.)

Errors marked in the Status column on the Rule Sheet can usually be corrected by editing the rule or correcting input values on the Variable Sheet.

Occasionally an error message is provided for a function that is called indirectly during solution.

For further information about valid arguments for TK!Solver built-in functions, see the Built-in Functions section.

The following explanatory error messages are provided:

› ACOS: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› ACOSH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› APPLY: Argument error

Cause: Supplying a first argument for the APPLY function that does not evaluate to a symbolic value; supplying a symbolic value or expression for a symbolic value that is not the name of a previously defined user function.

› ASIN: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› ASINH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› ATAN2: Argument error

Cause: Supplying a second argument value that is not possible with the specified function value in the first argument.

› ATANH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› Argument error

Cause: Supplying an argument value to a function that causes an unknown error.

› COSH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› COUNT: Argument error

Cause: Supplying a single argument for the COUNT function that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› DOT: Argument error

Cause: Supplying either argument to the DOT function that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; one of the named lists contains no values.

› Division by zero

Cause: Evaluating a rule or function necessitating division by zero, which is undefined.

› ELEMENT: Argument error

Cause: Supplying a first argument for the ELEMENT function that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values; supplying a value or expression for the second argument that does not evaluate to an integer, or is not a valid element number and no third argument is specified.

› EXP: Argument error

Cause: Supplying a value or expression that is equal to or greater than 1E120 as an argument to the EXP function.

› Exponent overflow

Cause: Evaluating a rule or function that results in a value equal to or greater than 1E120.

› Inconsistent

Cause: Attempting a solution for a rule that involves inconsistent variable values.

› Invalid symbolic computation

Cause: Attempting a solution for a rule or function in which a numeric calculation is attempted with a symbolic value.

› Iteration: Dependency error

Cause: Attempting a solution for a rule that is dependent on the solution of another rule. Solution of one rule implies solution of the other.

› Iteration: Out of memory

Cause: Attempting an Iterative solution that exhausts the remaining available memory.

› Iteration: Too many guesses

Cause: Attempting an Iterative solution that involves too many guess variables.

› LN: Argument error

Cause: Supplying a value or expression for a negative number or zero as an argument to the LN function.

› LOG: Argument error

Cause: Supplying a value or expression for a negative number or zero as an argument to the LOG function.

› MAX: Argument error

Cause: Supplying a single argument for the MAX function that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› MIN: Argument error

Cause: Supplying a single argument for the MIN function that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› MOD: Argument error

Cause: Supplying a value or expression as a second argument that evaluates to zero.

› NPV: Argument error

Cause: Supplying a first argument for the NPV function that does not evaluate numerically; supplying a single second argument that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› POLY: Argument error

Cause: Supplying a first argument for the POLY function that does not evaluate numerically; supplying a single second argument that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› SINH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› SQRT: Argument error

Cause: Supplying a value or expression for a negative number as an argument.

› SUM: Argument error

Cause: Supplying a single argument that is not a symbolic value or an expression that evaluates to a symbolic value for a list name; supplying a value or expression for a list name that contains no values.

› TANH: Argument error

Cause: Supplying an argument value that is not in the domain of the function.

› User function: Argument error

Cause: Supplying a value or expression as an argument to a user function that is not in the domain of the function.

The following system errors may also appear in the Status column on the Rule Sheet by attempting an action that causes an unknown program error. (These conditions should not occur.) In the case of such an error, save a copy of the model and a list of the actions that produced the error, note which error message was displayed, and contact your dealer.

› System error: evaluation error
› System error: stack overflow
› System error: stack underflow

Bottom of Screen:
The following messages appear on the bottom line of the screen when a hardware interface problem occurs for most implementations of the TK!Solver program:

DISK ERROR - Press ENTER to try again.

Cause: Attempting a print or storage action to: a disk drive that does not exist; an empty disk drive; or an unformatted, full, worn, write-protected, or damaged diskette.

Actions: Specifying a valid drive specifier, or inserting a usable diskette and pressing the Enter key; or pressing the Break key to return to the program and interrupt the disk action. Sometimes a more diagnostic message appears on the Prompt/Error line after pressing the Break key.

Insert program disk, then press ENTER.

Cause: Attempting an action that requires dynamic loading from a program diskette that has been removed from the disk drive.

Actions: Inserting the TK!Solver program diskette or a working diskette that contains the TK!Solver Overlay File and pressing the Enter key; or pressing the Break key to produce a confirmation message to cancel the program.

Press BREAK again to cancel the program.

Cause: Pressing the Break key in response to the message to insert a program diskette.

Actions: Pressing the Break key a second time to cancel the program, or any other key to repeat the message to insert a program diskette.

Press ENTER when printer is ready.

Cause: Attempting to output to a printer that is unavailable for some reason.

Actions: Verifying that a printing device is attached, on-line, not currently printing, supplied with paper, etc., and pressing the Enter key; or pressing the Break key to return to the program and interrupt the TK!Solver print action.

Value Locations:

The following message can appear anywhere program output values are displayed:

=ERROR=

Cause: Attempting to enter numbers in lists for a plot that generate display scales that are too large; attempting a calculation or solution with the program that results in a value that is outside the numeric limits for the program in an output or list value field.

Actions: Verifying that the difference between the smallest and largest values in lists used in plots has an absolute value less than 1E120. Generating a calculation or solution that has an absolute value less than 1E120; saving a copy of the model and a list of the actions that produced the error in a field, noting the error message above, and contacting your dealer.

Cleared Screen:
The following error message appears at the top of a cleared screen:

A NON-RECOVERABLE PROGRAM ERROR
has occurred. Please write down
the information listed below and
refer to the Reference manual
for further instructions.

Cause: Attempting an action that causes an unknown program error. (This condition should not occur.)

Actions: Writing down the numbers listed below the message, noting the error message above and the actions that may have produced the error, saving the diskette, and contacting your dealer.

Glossary

The following terminology is used in the documentation for the TK!Solver program:

Action An activity performed by the program in response to a user command.

Address The specification representing a field position on a sheet. An address may be represented by a row number, a character representing the last row, a label letter, or a combination of these. An address is used with the Destination option of the Goto command to move the cursor to a new position.

Argument A variable or expression representing a value in the domain for a built-in or user function.

Arithmetic Operator The special symbol used to represent an arithmetic operation in a rule or expression. The operators in the TK!Solver program include:

+ - / * ^

Arrow Keys Keys used to move the cursor around a sheet in Command mode or to move the cue in the Editor.

Associated List A list associated with a particular variable.

Automatic Iteration The process of setting up guess values so that if the Iterative Solver is required for a solution it can be called automatically.

Block A number of rows or column of fields in the program. A block is designated by the user by pointing to or specifying its beginning and ending points.

Booting Loading the diskette containing the computer's operating system.

Built-in Function A function provided by the TK!Solver program.

Byte A unit of measurement for computer memory. On most microcomputers, one byte equals one character. Memory is often approximated in K bytes (1 K byte = 1024 bytes).

Calculation Unit The unit of measurement in which a variable or list is first defined and calculations are performed.

Calculator The TK!Solver feature that instantly evaluates expressions entered into value entry fields, if all variables and arguments to functions have defined values.

Case Insensitive Interpreting uppercase and lowercase characters interchangeably. In the TK!Solver program, the names of built-in functions are case insensitive.

Case Sensitive Distinguishing between uppercase and lowercase characters. In the TK!Solver program, all names except built-in functions are case sensitive.

Character Any printable letter, digit, or special symbol produced by a single keystroke.

Column A vertical list of fields under a label in the Field Area of a sheet.

Command A keystroke or set of keystrokes that directs the program to perform some action.

Command Mode The mode in which the TK!Solver program accepts commands.

Comment A note users can enter in a field for their own reference or to display text. The TK!Solver program does not evaluate comments.

Comparison Tolerance The number defined on the Global Sheet used in establishing equality. Generally, two numbers are considered equal if their difference divided by the larger of the two numbers is less than the comparison tolerance.

Cue The marker that moves within text in an entry field or a user response on the Prompt/Error line before the Enter key is pressed.

Current The present status of the program, cursor, cue, or screen display.

Cursor The marker used to move around the Information Area and highlight a field.

Data Information that can be entered into the program.

Default The value supplied or action taken when no alternative has been specified.

Destination The position to which an item is moved or copied.

DIF File A file using the data interchange format standard developed by Software Arts, Inc. to permit exchange of data between programs.

Direct Solver The solver that solves by substituting known values to determine unknown values.

Directory Scrolling Successively displaying a selection of files from a diskette with an option of the Storage command so that one can be chosen for use.

Disk Drive The hardware component where the diskette is inserted, allowing the computer to read the contents of the diskette.

Diskette A plastic disk that stores computer programs magnetically.

Display Size The portion of the current item displayed in a field or on the Status line, or the part of a user response that fits in the display on the Prompt/Error line.

Display Unit The unit of measure in which the values for a variable or list are currently displayed.

Diving Accessing a subsheet from a related sheet or subsheet by using the Dive command.

Domain The permissible values for the argument or arguments of a function.

Drive Specifier The name used to designate the appropriate disk drive in file specifications.

Edit Indicator The word "Edit" in parentheses that appears on the Prompt/Error line when the user is using the Editor.

Editing Changing an item or user response by calling the Editor on the current entry field or on the Prompt/Error line.

Editor The facility used to edit items or user responses.

Element The sequential number for a value in a list.

Entering Putting an item in a field by pressing the Enter key or one of the Arrow keys, or completing a user response on the Prompt/Error line by pressing the Enter key.

Entry The characters considered valid for a particular field.

Entry Field A field that accepts characters entered with the Enter key or an Arrow key. Entry fields accept Name, Unit Name, Comment, Value, Numeric Value, Integer, Rule, Filename, or Setup String entries.

Entry Mode The mode in which new information is typed and entered into entry fields.

Equation Two expressions related by an equal sign.

Error Condition A mistake or inconsistency recognized by the program and indicated by an error signal and an error message on the Prompt/Error line, an error symbol in a Status field on the Variable and Rule Sheets, a question mark in a field for a unit error, or a message on the bottom of the screen for a hardware error.

Error Message A message displayed by the program whenever an attempt is made to enter an invalid entry or option, or perform an invalid action.

Evaluation Calculation of the value of an expression by the Calculator, or of variables in an equation by one of the solvers.

Expression One or more characters that can be evaluated, containing numeric values, symbolic values, variable names, arithmetic operators, or function names in any valid combination.

Extended Command A command that requires more than one keystroke, beginning with one of the following characters:

/ = : ? !

Extension Characters following a delimiter in a file specification, often indicating the type of file.

Field A position on a sheet that contains items.

Field Area The sheet area containing fields in rows under column labels in which specific information for a sheet can be entered.

Field Type The quality of a field that determines what can be entered into it and the actions that can be performed on it. The TK!Solver program has three kinds of fields: entry, option, and output.

File	A collection of data on a diskette in which information or programs can be stored. A number of files can usually be stored on the same diskette.
File Specification	The full name of a file, dependent on the conventions for the computer system being used, including such parts as a disk specifier, filename, and file extension.
Filename	The name used to identify a particular file.
Formatting	The process performed on the operating system level that prepares diskettes to accept data for the first time.
Function	A relationship between two sets of values, called the domain and range. Each element in the domain is paired with a unique element in the range. The TK!Solver program uses both built-in and user-defined functions.
Global Setting	A pre-defined and changeable value in a field on the Global Sheet that affects automatic insertion of variables, the solvers, or use of a printer or output file.
Guess	An estimated value for a variable used with the Iterative Solver.
Hardware	The physical components of the computer system being used, which might include a video screen, memory unit, keyboard, disk drives, a printer, and diskettes.
Header Area	The sheet area containing general information about the contents or use of a sheet. It consists of a block of rows, each containing one label and one field.
Help Facility	The TK!Solver facility accessed by typing ? that describes specific topics related to use of the program.
Information Area	The part of the screen below the Message Area where sheets, subsheets, and the Help Facility are displayed.
Initial Cursor Position	The position of the cursor the first time a sheet is accessed.
Input	Data entered by the user or loaded from a file.

Invalid An attempted command, action, user response, or entry that is not recognized by the program as correct for a particular location or at a particular time.

Item The valid contents of a field.

Item Size The complete contents of an item, regardless of what is actually displayed on the screen.

Iteration A method of problem solution that uses successive approximations of the answer to converge to a desired mathematical result.

Iterative Solver The solver that starts with a guess value for a variable, manipulates the guess to approximate the final value more closely, and continues this process until a solution is reached.

Label A heading that describes a field or column of fields.

Label Letter The letter used in field addresses to represent a label.

Label Line A line of column labels near the top of all sheets (except the Global Sheet and the Variable subsheets) describing the contents of the fields below them.

List A set of numbered values.

List Solver The solver that calls the Direct or Iterative Solver for lists of values associated with one or more variables.

Loading Copying data from a file on a diskette into the internal memory of the computer.

Memory The computer component that stores and retrieves information.

Memory Indicator A number in the righthand corner of the Message Area that indicates amount of memory remaining, expressed in thousands of bytes (K). When "LOW" is displayed, less than 1K remains; when "OUT" is displayed, there is only enough memory to save information, unless something is deleted.

Message Area The top two lines of the screen used by the program to display status information, prompts, and error messages, and to accept user responses.

Mode A program environment in which particular user actions are valid. The TK!Solver program has three modes: Command, Entry, and Option.

Model All data entered into the TK!Solver program necessary to express a particular class of problem.

Name The symbol used to refer to a variable, a list, or a user function in a name entry field, in an expression in a value field, or in a rule.

Operating System The program or series of programs that use the hardware facilities of the computer to run programs and perform actions.

Operating System Level The computer level at which the operating system works to perform such activities as copying diskettes and loading programs.

Option A single character that is valid for the current option field.

Option Field A field that accepts only one of a specific set of choices and does not require use of the Enter key.

Option Mode The mode in which only particular options are valid. The user is automatically placed in Option mode on option fields.

Output Information produced by a program.

Output Field A field that displays information generated by the program.

Plot A graph of list values defined on the Plot Sheet and displayed with the Action command.

Pointer Indicator A number and letter (or letter only) at the end of certain prompts that displays the current cursor position during pointing.

Pointing Moving the cursor to a final cursor position in response to a command prompt. The Blank, Copy, Delete, Move, Print, and DIF Save commands all use pointing.

Position Indicator The display of the current cursor position in the upper lefthand corner of the screen.

Precedence The order in which an expression or rule is evaluated, which can be altered by using parentheses.

Print File A file that contains the output from a plot or table, or a copy of a sheet or portion of a sheet in the program.

Program	A set of sequential instructions that direct the computer to accomplish a specific task.
Program Environment	The context in which actions take place. The TK!Solver includes Command, Entry, and Option modes, the Editor, and the Help Facility.
Prompt	A message generated by the program requesting a user response.
Prompt/Error Line	The line that displays prompts and error messages. It is the second line in the Message Area.
Random Access Memory	Computer memory that allows data to be stored and retrieved.
Range	The possible function values.
Row	A horizontal line of fields describing a particular entity in a Field Area, or a label and a field in the Header Area.
Rule	A statement in mathematical notation about the relationships between variables. In the TK!Solver program, all valid rules are equations.
Running	Causing the computer to start following a set of instructions to perform an activity or execute a program.
Satisfied	The condition of a rule when it has been solved.
Screen	The video display portion of the computer.
Scrolling	Changing the view of a sheet or field. The view of a sheet is scrolled by moving the cursor past the top or bottom of the current display. The view of a field is scrolled by typing or moving the cue past the right or left edge of the current display field.
Search Item	A specification of the exact displayed contents of a field in order to move to it. A search item is preceded by a quotation mark (”).
Series	Two or more expressions, separated by commas, used as arguments to a built-in function.
Setup String	A set of control codes and characters used to direct a printing device.
Sheet	A location for entering data and defining models. In the TK!Solver program, there are a variety of sheets containing different kinds of information.

Sheet Letter A letter representing a sheet name that is used to access that sheet.

Sheet Title Line The line at the top of a window that identifies the sheet being displayed.

Slash Command An extended command beginning with a slash (/).

Solution The process of solving a model.

Solution Indicator An exclamation point at the extreme right of the Message Area that is visible when equations are not satisfied and disappears when solution of any equation is accomplished.

Solver A TK!Solver feature for solving models. There are two solvers in the TK!Solver program: Direct and Iterative. The List Solver calls the Direct or Iterative Solver for list values associated with variables.

Special Keys Computer keys that perform actions and do not display characters. They include the Arrow keys, the Backspace key, the Break key, and the Enter key.

Startup The process of loading the TK!Solver program.

Status The current state or condition of the program or a model. Program status is displayed in the Message Area at the top of the screen; solution status is displayed in the Status fields of the Rule and Variable sheets.

Status Line The line at the top of the screen that displays cursor position, field contents, amount of remaining memory, and solution status of the model.

Status Message A message that displays on the Prompt/Error line during an action that takes time to perform.

Storage File A file that contains information that can be saved from the TK!Solver program or loaded into it. Storage files can be created by the user or purchased as part of a TK!SolverPack™.

Storage Unit The unit of measurement a list is stored in and calculations performed in.

Subsheet A sheet that provides detailed information about a defined variable, list or user function. Accessed by diving from a sheet.

Symbolic Value A non-numeric value. In the TK!Solver program, a symbolic value is preceded by an apostrophe (').

Table A display of list values defined on the Table Sheet and produced with the Action command.

Typical Value The value which, when multiplied by the comparison tolerance, determines at what point values will be considered near zero and the typical value will replace the comparison tolerance for determining equality.

Typing Pressing keys to input options, entries, or commands.

Unit The unit of measure the value of a variable or list is expressed in.

Unit Conversion The process of converting values from one unit of measurement to another.

Unsatisfied The condition of a rule when it has not been solved.

User Function A function defined by the user on the User Function Sheet and subsheets that takes one argument.

User Response Something typed by the user on the Prompt/Error line in response to a prompt requesting information and ending in a colon.

Valid A command, user response, or item that is recognized by the program as correct for a particular location or at a particular time.

Value The assigned or calculated quantity of a variable. In the TK!Solver program, values can be expressed as real or integer numbers, variables, symbolic values, or expressions.

Variable An entity that can take on different values.

View The part of a sheet or subsheet currently displayed on the screen.

Wildcard A character specific to an operating system that can be substituted for one or more characters when specifying an unknown filename or a series of filenames.

Window The portion of the screen through which sheets, subsheets, and the Help Facility are displayed. The TK!Solver screen can be divided into two windows at one time.

Index

⇐ (key) iii; III-3; VI
⇒ (key) iii; III-3; VI
⇑ (key) iii; III-3; VI
⇓ (key) iii; III-3; VI
♦ (key) iii; III-4; VI
⊗ (key) iii; III-5; VI
◄ (key) iii; III-5; VI

: (command) III-23
= (command) III-42
> (command) III-16
< (command) III-41
; (command) III-51
◪ (command) iii; III-18; VI
? (command) I-15; III-26
/ (commands) I-12; III-2; A-7
/! (command) III-44; IV-7
! (command) III-6; IV-7
! (solution indicator) I-4, 23; A-17

ABS function V-1
absolute value function V-1
ACOS function V-1
ACOSH function V-1
Action command III-6; IV-7
accuracy I-8; IV-2, 24
addition IV-3,6
Add Offset field II-15
address I-3, 7; III-23
algebraic expressions IV-4
All option, Reset command III-40
Alphabetic Cross-reference v
alphanumeric characters I-17
APPLY function V-2
arc cosine function V-1
arc hyperbolic cosine function V-1
arc sine function V-2
arc hyperbolic sine function V-2
arc tangent functions V-2,3
arc hyperbolic tangent function V-2
argument IV-4, 8; V-1
argument errors A-45
arithmetic operators IV-3
Arrow keys iii; III-3; VI
ASIN function V-2
ASINH function V-2
Associated List field II-9
associated lists II-4, 9; IV-19
ATAN function V-2
ATANH function V-2
ATAN2 function IV-7; V-3
automatic item insertion I-16; II-17, 24, 36
automatic iteration II-9, 17; IV-11,15
available memory I-3; VI; A-17, 41

/b (command) III-10
Backspace key iii; III-5; VI
backup diskettes VI
base e logarithm function V-4
Blank command III-10
blank diskettes VI
Blank option, Variable Sheet II-5
blanking fields III-10
blanking values II-5
Block option, List command III-32; IV-20
boldface type iii
booting the system VI
Break key iii; III-5; VI
built-in functions IV-14, 24; Part V
bytes I-3; VI; A-17

/c (command) III-12
calculation I-8; IV-6
calculation errors I-23
Calculation units II-6, 10, 13; IV-16
calculator IV-6
calling a sheet III-42
cancelling an entry or action III-5
caret character II-21; VI
Carriage Return/Linefeed Option II-22; VI
changing a sheet III-42, 51
Character field II-41; III-11
characters I-17; VI
clearing sheets III-39
column labels I-8
columns I-8
command errors I-22
Command mode I-12
command options I-4
commands v; I-12; Part III
comments I-18; II-7; IV-13
common logarithm V-5
comparison tolerance II-18, IV-8
computation I-8; IV-6
computer error signal I-5; VI
control character II-21
control codes I-20; II-21
Control key VI
conversion of units I-17; II-13
Copy command III-12
copying a field III-12
COS function V-3
COSH function V-3
cosine functions V-1,3
COUNT function IV-14; V-3
count list function IV-14; V-3
Cross-references v, viii
cue I-10
current field I-10
cursor I-10
/d (command) III-14
default settings II-16; VI
Delete command III-14
Delete File option, Storage command III-49
deleted entities A-23
deleting rows III-14
destination III-23
device names I-19; VI
DIF clearinghouse A-3
DIF format files IV-18; A-2
DIF storage options III-47
Direct Solver III-44; IV-7,19
directory scrolling III-3, 45; VI
disk drives VI
disk errors A-22, 24
disk operating system VI
disk specifiers VI
diskettes VI
Display Scale ON field II-40
display size I-3, 7, 8, 11; IV-2
Display units I-18; II-6, 10; IV-16
displaying sheets III-39, 52
Dive command III-16
division IV-3, 6
division by zero IV-8
domain II-30, 31; IV-22
domain lists II-31; IV-19
DOT function IV-14; V-3
Down Arrow key iii; III-3; VI
drive specifiers VI

/e (command) I-14; III-21
e (base of natural logarithm) V-4
e (scientific notation) IV-1
E function V-4
Edit command I-14; III-18
Edit Field command I-14; III-21
Edit Indicator I-4

editing a rule III-21; IV-14
editing a user response III-18
editing an item I-9,13; III-18, 21
Editor I-14
element numbers III-32; IV-20
Element fields II-24, 28, 36
ELEMENT function IV-14; V-4
Enter key iii; III-4; VI
entering data I-13
entry errors I-22
Entry fields I-16
Entry mode I-13
equal sign IV-5
equality IV-8
equation components IV-1
equations I-19; II-11; IV-5
erasing data III-10, 14, 38, 39
error conditions I-22
error messages vii; I-5, 22; A-22
error signal I-5; VI
escape character I-20; II-21
evaluation IV-6, 7, 19
exclamation point I-4
exiting the Editor I-14
exiting the Help Facility I-15; III-26
exiting the program I-1; III-38
EXP function V-4
exponentiation IV-1, 3, 6
expression evaluation IV-6
expressions IV-4; V-1
extended commands III-2
extensions of filenames VI

factors IV-1
Field Area I-8
Field Items I-3, 8
Field Labels I-3, 8
field types I-16
fields I-7, 16
file extensions VI
file manipulation III-45; A-1
filenames I-19; II-20; VI
files VI; A-1
Fill Action, Action command III-7
filling in a list III-7; IV-19
final cursor position III-2
first guesses II-9; IV-11
Form Length field II-19
formats for printing II-18
formatting diskettes VI
From field II-14
function argument errors A-45
function arguments IV-4; V-1
function inverses IV-8
Function Save option, Storage command III-47
Functional Cross-reference viii
functions IV-4, 22
functions, built-in IV-24; Part V
functions, user-defined II-30, 34; IV-22

general program commands ix
Get option, List command III-33; IV-21
GIVEN function V-4
global settings II-16; VI
Global Sheet II-6; IV-15
going to a field I-10; III-3, 23
going to a sheet III-42, 51, 52
going to a subsheet III-16
Goto command III-23
graphs II-38; III-8
guess values II-5, 9; IV-10, 11

Hardware Reference Part VI
hardware requirements VI
Header Area I-7
Header Field I-7
Help command III-26
Help Facility I-15
Help File VI
hexadecimal codes II-21; VI
Horizontal option, Table Sheet II-44
hyperbolic cosine function V-3
hyperbolic sine function V-7
hyperbolic tangent function V-8

/i (command) III-29
inconsistency errors I-23; A-44
indirect unit conversion II-13; IV-17
indeterminable values IV-7
Information Area I-5
initial cursor position I-10
initial guess II-9; IV-11
input lists III-30; IV-19
Input option, Variable Sheet II-4
input values II-4, 6, 10, 28, 36
inputting data I-13, 16
Insert command III-29
inserting a row III-29
inserting variables I-16; II-6, 17
Instruction Files VI
insufficient memory I-3; VI; A-17
INT function V-5
integer entries I-19
integer truncation function V-5
intermediate redisplay II-17; IV-10
internal representation of numbers I-16; IV-2
interpolation II-31; IV-22
invalid entries I-16, 22
invalid program disk VI
inverse of functions IV-8
irregular interpolation IV-23
italic typeface iii
item length I-8
items I-8
iteration number II-18
Iterative Solver III-6; IV-10

k (bytes) I-3; VI
keyboards iii
keys, computer specific iii; VI

/l (command) III-30
label letters I-3; II-2
labels I-7
least value in list V-6
Left Arrow key iii; III-3; VI
Left Margin field II-20
Line End character II-22; VI
linear interpolation II-31; IV-22
list association II-4, 9; IV-14
list block solution III-32; IV-20
list creation II-9, 23, 26; IV-19
List command III-30; IV-19
list count function V-3
list element function V-4
list field II-44
List Fill-in III-7; IV-19
list functions IV-14; V-3,4,5,6, 8
list names II-23
List option, Variable Sheet II-4
List Sheet II-23
list storage III-47
List solver III-30; IV-11, 19
list subsheets II-24

list units IV-17, 19
list value assignment II-28, 36; III-7
literals I-18; IV-2; A-2
LN function V-5
loading a file III-48; VI
LOG function V-5
logarithm, base e V-5
logarithm, natural V-5
LOW (memory) I-3; A-17
lowercase iii

/m (command) III-34
machine-specific iii; I-1, 8; VI
making changes I-14
manual conventions iii
manual summary i
mapping II-31; IV-22
mathematical functions IV-22; Part V
mathematics Part IV
max function IV-14; V-5
maximum integer value I-19; VI
maximum item length I-16
maximum iteration count II-18; IV-11
maximum numeric value IV-2
maximum value function V-5
memory I-3; VI; A-17, 38
Memory Indicator I-3
Message Area I-2
MIN function IV-14; V-6
minimum integer value I-18; VI
minimum numeric value IV-2
minimum value function V-6
MOD function V-6
model development IV-13, 14, 16
model reset III-39
model solution I-23; IV-7
modes I-12
modification commands viii
modulus function V-6
Move command III-34
movement I-10
movement commands viii
moving a row III-34
multiplication IV-3,6
Multiply By field II-14

name entries I-17
natural logarithm function V-5
negative sign IV-6
net present value function V-6
NPV function V-6
numbers IV-1
numeric value entries I-19

OCR-B typeface iii
One-Window option, Window Command III-53
operating system VI
operators IV-3
option errors I-22
Option field I-20
option items I-8
Option mode I-14
OUT (memory) I-3; VI; A-17
output device names II-20
output errors I-23
output fields I-23
output items I-8
output lists III-30; IV-19
Output option, Variable Sheet II-4
outputting to a file II-20; III-46; VI; A-1
overdefined variables A-44
Overlay File VI
overview Part I
overwriting a file III-46
overwriting during loading III-48

/p (command) III-36
Page Breaks ON field II-18
Page Numbers ON field II-19
PI function V-6
Plot Action, Action command III-8
Plot Files A-1
Plot Sheet II-38
plotting II-38; III-8; IV-15
Pointer Indicator I-4
pointing to a field I-4; III-2
POLY function V-7
polynomial function V-7
polynomials IV-11
Position Indicator I-3
positive sign IV-6
precedence IV-6
precision IV-2, 8
Print command III-36
Print Files II-18; A-1
Printed Page Length field II-20
Printed Page Width field II-20
printers II-20; VI
Printer Device Or Filename field II-20
printer setup strings I-20; II-21; VI
printing II-39, 43; III-8, 36
program environments I-12
program errors I-22; VI; A-22,52
program overview Part I
program startup I-1; VI
prompts I-4; A-7
Prompt/Error line I-4
Put option, List command III-32
putting values into lists III-7, 32; IV-20

/q (command) III-38
question mark I-23; II-6
Quit command III-38
quitting the Help Facility I-15
quitting the Editor I-14
quitting the program III-38

/r (command) III-39
range II-30, 32; IV-22
range lists II-36
range of integers I-17, 19
range of numbers IV-2
refreshing the screen VI
remaining memory I-3; VI; A-17
Reset command III-39
restoring an item I-13
Return command III-41
Right Arrow key iii; III-3; VI
rounding off numbers IV-2
rows I-8
rule errors I-23
Rule field II-12
Rule Sheet II-11
rules IV-5, 7
running the program VI

/s (command) III-45
satisfied rules I-23
save options, Storage command III-46
saving files III-46
scientific notation IV-1
screen I-2; VI
Screen Or Printer fields II-39, 43
scrolling through a field I-11
scrolling through a sheet I-10
scrolling through filenames III-3, 45; VI
Search option, Goto command III-24

Select command III-42
selecting a filename III-45
selecting a sheet III-42
series V-1
setup string entries I-20; VI
SGN function V-7
Sheet option, Reset command III-39
Sheet Title line I-7
sheets vii, x; I-5; Part II; IV-14
sign function V-7
significant digits IV-2
signum function V-7
SIN function V-7
single-rule evaluation IV-7
SINH function V-7
size constraints VI
slash commands A-7
solution IV-7
solution commands ix
solution errors I-23; IV-7
Solution Indicator I-4, 23
Solve action, Action command III-6
Solve command III-44
Solve option, List command III-30; IV-19
solvers III-6, 30; IV-7
special keys iii; III-1; VI
splitting windows III-52
SQRT function V-8
square root function V-8
startup I-1
Startup File VI
status fields I-20; II-4, 9
Status line I-1,3, 5
status messages I-5; A-17
STEP function V-8
step interpolation II-31; IV-23
stopping an action or command I-22; III-5
stopping the program III-38
Storage command III-45
storage diskettes VI
Storage Files iv; A-2
Storage Unit field II-28
Storage units II-24; IV-17
strings VI
subsheets vii, x; I-6; II-2
subtraction IV-3, 6
SUM function IV-14; V-8
Switch command III-51
symbolic values I-18; IV-2; V-1; A-2
symbols VI
system level VI
system notes VI

Table action, Action command III-8
Table Files A-1
table lookup II-32; IV-22
Table Sheet II-42; IV-15
TAN function V-8
TANH function V-8
terminology Glossary
Title fields II-39, 43
TK!Solver files A-1
TK!Solver functions IV-24; Part V
TK!Solver Help File VI
TK!Solver Overlay File VI
TK!Solver program diskettes VI
TK!Solver screen I-2; VI
TK!Solver Startup File VI
TK!SolverPack™ VI
To field II-14
trigonometric functions IV-24; Part V
Two-window option, Window command III-53
typical value II-18; IV-9

unary plus and minus signs IV-6
unevaluated expression errors A-31, 44

unit abbreviations A-4
unit conversions II-7, 13; IV-16
unit errors I-22
unit fields II-6, 22
unit name conventions A-4
unit name entries I-17
Unit Save option, Storage command III-47
Unit Sheet II-13
units of measurement I-17; II-7, 13; IV-16
unsatisfied rules I-23; II-11
Up Arrow key iii; III-3; VI
uppercase iii
user function names I-17
User Function Sheet II-30
User Function subsheets II-35
user functions II-30, 35; IV-22
user responses I-4

valid expressions IV-4
valid filenames I-19; VI
valid items I-16
value entries I-18
Variable Insert ON field II-17
variable names I-17
Variable Save option, Storage command III-47
Variable Sheet II-3
Variable subsheets II-8
Variable option, Reset command III-39
variables I-16; II-3; IV-3
Vertical option, Table Sheet II-44
Vertical Or Horizontal field II-44
video monitors iv, VI

/w (command) III-52
wildcards III-45; VI
Window command III-52
windows I-5

x (variable name) V-1
X-axis field II-40

Y-axis field II-40

zero I-17; IV-2, 7, 9